

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization International Bureau



(43) International Publication Date
17 June 2004 (17.06.2004)

PCT

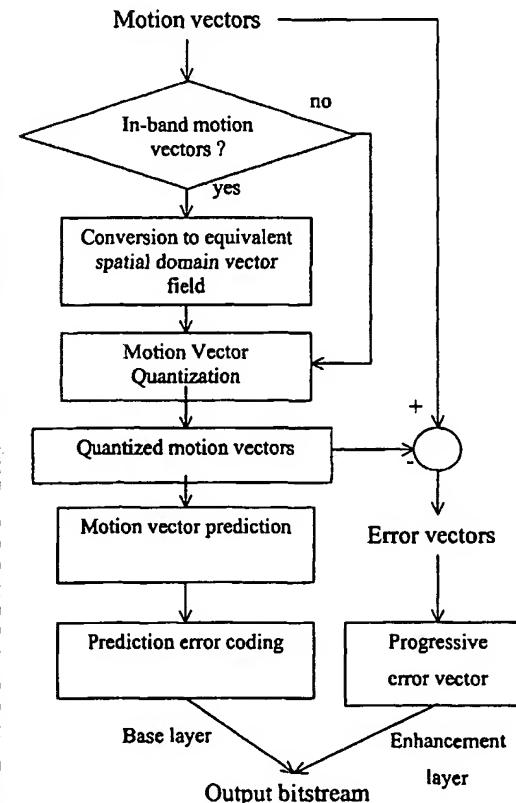
(10) International Publication Number
WO 2004/052000 A2

- (51) International Patent Classification⁷: H04N 7/00
- (21) International Application Number: PCT/BE2003/000210
- (22) International Filing Date: 4 December 2003 (04.12.2003)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
0228281.2 4 December 2002 (04.12.2002) GB
- (71) Applicants (*for all designated States except US*): INTERUNIVERSITAIR MICROELEKTRONICA CENTRUM [BE/BE]; Kapeldreef 75, B-3001 Leuven (BE). VRIJE UNIVERSITEIT BRUSSEL [BE/BE]; Pleinlaan 2, B-1050 Brussels (BE).
- (72) Inventors; and
- (75) Inventors/Applicants (*for US only*): BARBARIEN, Joeri [BE/BE]; C. Serneelsstraat 24, B-2222 Itegem (BE). MUNTEANU, Adrian [RO/BE]; Kroonlaan 327, B-1050 Ixelles (BE). ANDREOPOLUS, Ioannis [GR/BE]; Rond-Point de l'Etoile, 5/7eme étage, B-1050 Ixelles (BE). SCHELKENS, Peter [BE/BE]; Eduard Anseelestraat 120, B-2830 Willebroek (BE).
- (74) Agents: BIRD, William, E. et al.; Bird Goen & Co, Klein Dalenstraat 42A, B-3020 Winksele (BE).
- (81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.
- (84) Designated States (*regional*): ARIPO patent (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),

[Continued on next page]

(54) Title: METHODS AND APPARATUS FOR CODING OF MOTION VECTORS

WO 2004/052000 A2



(57) Abstract: A method and apparatus is described for coding motion information in video processing of a stream of image frames and for avoiding the drift problem. The method or apparatus is for providing motion vectors of at least one image frame, and for coding the motion vectors to generate a quality-scalable representation of the motion vectors. The quality-scalable representation of motion vectors and a set of one or more enhancement-layers of motion vectors. The method of decoding and a decoder for such coded motion vectors as part of receiving and processing a bit stream at a receiver includes the base-layer of motion vectors being losslessly decoded, while the one or more enhancement layers of motion vectors are progressively received and decoded, optionally including progressive refinement of the motion vectors, eventually up to their lossless reconstruction.



Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),
European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE,
ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE,
SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA,
GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declaration under Rule 4.17:

- *of inventorship (Rule 4.17(iv)) for US only*

Published:

- *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

METHODS AND APPARATUS FOR CODING OF MOTION VECTORS**Field of the invention**

The invention relates to methods and apparatus and systems for coding framed data especially methods, apparatus and systems for video coding, in particular those exploiting subband transforms, in particular wavelet transforms. In particular the invention relates to methods and apparatus and systems for motion vector coding of a sequence of frames of framed data, especially methods, apparatus and systems for motion vector coding of video sequences, in particular those exploiting subband transforms, in particular wavelet transforms.

Background of the invention

Video codecs are summarised in the book "Video coding" by M. Ghanbari, IEE press 1999. A basic method of compressing video images and thus to reduce the bandwidth required to transmit them is to work with differences between images or blocks of images rather than with the complete images themselves. The received image is then constructed by assembling later images from a complete initial image modified by error information for each image. This can be extended to determining motion of parts of the image – the motion can be represented by motion vectors. By making use of the error and motion vector information, each frame of the received image can be reconstructed. The concept of scalability is introduced in section 7.5 of the above book. Ideally the transmitted bit stream is so organised that a video of preferred quality can be selected by selecting a part of the bit stream. This may be achieved by a hierarchical bit stream, that is a bit stream in which the data required for each level of quality can be isolated from other levels of quality. This provides network scalability, i.e. the ability of a node of a network to select the quality level of choice by simply selecting a part of the bit stream. This avoids the need to decode and re-encode the bit-stream. Such a hierarchically organised bit stream may include a "base layer" and "enhanced layers", wherein the base layer contains the data for one quality level and the enhanced layer includes the residual information necessary to enhance the quality of the received image. Preferably, the type of scalability, e.g. spatial or temporal can be selected independently of each other, i.e. different types of scalability are supported by the same data stream – this is called hybrid scalability.

Certain transforms have been used to assist in video compression, e.g. the

discrete wavelet transform (DWT), see for example: "Wavelets and Subbands", A. Abbate et al., Birkhäuser, 2002. Wavelet video codecs based on spatial-domain MCTF (SDMCTF) are presented in D. S. Turaga and M. v d Schaar, "Unconstrained motion compensated temporal filtering," ISO/IEC JTC1/SC29/WG11, m8388, MPEG meeting, 5 Fairfax, USA, May 2002, B. Pesquet-Popescu and V. Bottreau, "Three-dimensional lifting schemes for motion compensated video compression," *Proc. IEEE ICASSP*, Salt Lake City, UT, May 7-11, vol. 3, pp. 1793 -1796, 2001, J.-R. Ohm, "Complexity and Delay Analysis of MCTF Interframe Wavelet Structures," ISO/IEC JTC1/SC29/WG11, m8520, MPEG-meeting Klagenfurt, July 2002, and Y. Zhan, M. 10 Picard, B. Pesquet-Popescu and H. Heijmans, "Long temporal filters in lifting schemes for scalable video coding," ISO/IEC JTC1/SC29/WG11, m8680, MPEG meeting, Klagenfurt, July 2002. In these schemes, the motion estimation and compensation (ME/MC) are performed in the spatial domain. Afterwards, the prediction errors are wavelet transformed and the transform coefficients are entropy coded.

15 It is also possible to perform the motion compensation and estimation in the transformed domain. Coding of the transformed image is called in-band coding. Because the motion estimation is performed in the wavelet domain, each resolution level has a set of motion vectors associated to it. This may have the disadvantage that the number of motion vectors increases because of the increased number of levels of 20 representation. The final bit stream, which is a combination of error images and motion vectors, then requires more bandwidth. Ideally, to avoid a performance penalty when decoding to lower resolutions, only the motion vector data associated with the transmitted resolution levels should be sent. Hence, the system used to encode the motion vector data has to take this into account and has to produce a resolution 25 scalable bit-stream.

Summary of the invention

The present invention provides in one aspect a method of coding motion information in video processing of a stream of image frames, comprising:

30 providing motion vectors for at least one image frame,
quantizing the motion vectors to generate a set of quantized motion vectors equivalent to the motion vectors,
compressing the quantized motion vectors losslessly,
generating error vectors, each error vector being a difference between a motion vector

and its quantized equivalent, and
progressively encoding the error vectors in a lossy-to-lossless manner.

The present invention also provides a method of decoding encoded motion vectors in a bitstream received at a receiver and coded by the above method, the
5 decoding method comprising progressively decoding the error vectors in a lossy-to-lossless manner.

The present invention also provides a method of providing a representation of motion information in video processing of a stream of image frames, comprising:
providing in-band motion vectors of at least one image frame,
10 converting the in-band motion vectors to a spatial domain to generate motion vectors equivalent to the in-band motion vectors,
non-linearly predicting prediction motion vectors from spatial correlation of neighbouring motion vectors in one image frame,
generating prediction-error vectors from differences between the motion vectors in the
15 spatial domain and the prediction motion vectors,
coding the prediction error vectors, and
outputting the coded prediction-error vectors.

The present invention also provides a method of decoding encoded motion vectors in a bitstream received at a receiver having been encoded by the above
20 method, the decoding method comprising progressively decoding the coded prediction error vectors.

The present invention provides a method of providing a representation of motion information in video processing of a stream of image frames, comprising:
providing in-band motion vectors of at least one image frame,
25 converting the in-band motion vectors to a spatial domain to generate motion vectors equivalent to the in-band motion vectors,
transforming the motion vectors in the spatial domain to a wavelet domain using an integer wavelet transform to generate wavelet coefficients, and
coding the wavelet coefficients.

30 The present invention also provides a method of decoding a bitstream received at a receiver which has been coded by the above method, the decoding method comprising decoding the wavelet coefficients and generating the motion vectors.

The present invention provides a method of coding motion vectors of at least one image frame in video processing of a stream of image frames, comprising:

transforming the motion vectors using the integer wavelet transform to generate wavelet coefficients, and coding the wavelet coefficients.

The present invention provides a method of decoding a bitstream received at a receiver which has been coded by the above method, the decoding method comprising decoding the wavelet coefficients and generating motion vectors from the decoded wavelet coefficients.

The present invention provides a method of coding motion information in video processing of a stream of image frames, comprising:

10 providing motion vectors of at least one image frame, and coding of the motion vectors to generate a quality-scalable representation of the motion vectors.

The present invention also provides a method of decoding a bitstream received at a receiver which has been coded by the above method, the decoding method comprising decoding a base layer of motion vectors and an enhancement layer of motion vectors and enhancing a quality of a decoded image by improving the quality of the base layer of motion vectors using the enhancement layer of motion vectors.

The present invention also provides an encoder for coding motion information in video processing of a stream of image frames, comprising:

20 means for providing motion vectors for at least one image frame, means for quantizing the motion vectors to generate a set of quantized motion vectors equivalent to the motion vectors, means for compressing the quantized motion vectors losslessly, means for generating error vectors, each error vector being a difference between a motion vector and its quantized equivalent, and means for progressively encoding the error vectors in a lossy-to-lossless manner.

The present invention also provides a device for providing a representation of motion information in video processing of a stream of image frames, comprising:

means for providing in-band motion vectors of at least one image frame,

30 means for converting the in-band motion vectors to a spatial domain to generate motion vectors equivalent to the in-band motion vectors, means for non-linearly predicting prediction motion vectors from spatial correlation of neighbouring motion vectors in one image frame, means for generating prediction-error vectors from differences between the motion

vectors in the spatial domain and the prediction motion vectors,
means for coding the prediction error vectors, and
means for outputting the coded prediction-error vectors.

- The present invention also provides a device for providing a representation of
5 motion information in video processing of a stream of image frames, comprising:
means for providing in-band motion vectors of at least one image frame,
means for converting the in-band motion vectors to a spatial domain to generate
motion vectors equivalent to the in-band motion vectors,
means for transforming the motion vectors in the spatial domain to a wavelet domain
10 using an integer wavelet transform to generate wavelet coefficients, and
means for coding the wavelet coefficients.

- The present invention also provides an encoder for coding motion vectors of at
least one image frame in video processing of a stream of image frames, comprising:
means for transforming the motion vectors using the integer wavelet transform to
15 generate wavelet coefficients, and
means for coding the wavelet coefficients.

- The present invention also provides an encoder for coding motion information
in video processing of a stream of image frames, comprising:
means for providing motion vectors of at least one image frame, and
20 means for coding of the motion vectors to generate a quality-scalable representation of
the motion vectors.

- The present invention also provides a decoder for all of the encoders above.
The present invention also provides a computer program product which when
executed on a processing device executes any of the methods of the present invention.
25 The present invention also provides a machine readable data carrier storing the
computer program product .

BRIEF DESCRIPTION OF THE DRAWINGS

- 30 Figures 1a-c show general setups of coders for spatial (1a), in-band (1b) and hybrid
(1c) video codecs using either spatial or in-band motion estimation or in-band motion
estimation based on the CODWT.

Figure 2 shows per-level in-band motion estimation and compensation in accordance
with an embodiment of the present invention.

Figure 3 shows a layout of the motion vector set produced by in-band motion estimation in accordance with an embodiment of the present invention.

Figures 4a, b show flow diagrams of motion vector coding techniques in accordance with embodiments of the present invention.

5 Figure 5 shows neighboring motion vectors involved in the prediction in accordance with an embodiment of the present invention.

Figure 6 shows motion vectors used to predict in prediction scheme 2 in accordance with an embodiment of the present invention.

Figure 7 shows prediction scheme 3 in accordance with an embodiment of the present
10 invention.

Figure 8 shows prediction scheme 4 in accordance with an embodiment of the present invention.

Figure 9 shows examples of the two sets of flags transmitted by the prediction-error coder 3 in accordance with an embodiment of the present invention.

15 Figure 10 shows a 3D structure assembled in prediction error coder 5 in accordance with an embodiment of the present invention.

Figure 11 shows a structure of the motion vector set in accordance with an embodiment of the present invention.

Figure 12a shows a coder in accordance with a further embodiment of the present
20 invention.

Fig. 12b shows a flow diagram of motion vector coding techniques in accordance with a further embodiment of the present invention.

Fig. 13 shows a schematic representation of a telecommunications system to which any of the embodiments of the present invention may be applied.

25 Fig. 14 shows a circuit suitable for motion vector coding or decoding in accordance with any of the embodiments of the present invention.

Fig. 15 shows a further circuit suitable for motion vector coding or decoding in accordance with any of the embodiments of the present invention.

30 **DEFINITIONS**

Drift-free refers to the fact that both the encoder and decoder use only information that is commonly available to both the encoder and the decoder for any target bit-rate or compression ratio. With non-drift-free algorithms the decoding errors will propagate and increase with time so that quality of the decoded video decreases.

Resolution scalability refers to the ability to decode the input bit stream of an image at different resolutions at the receiver.

5 Resolution scalable decoding of the motion vectors refers to the capability of decoding different resolutions by only decoding selected parts of the input coded motion vector field. Motion vector fields generated by an in-band video coding architecture are coded in a resolution-scalable manner.

Temporal scalability refers to the ability to change the frame rate to number of frames ratio in a bit stream of framed digital data.

10 Quality of motion vectors is defined as the accuracy of the motion vectors, i.e. how closely they represent the real motion of part of an image.

Quality scalable motion vectors refers to the ability to progressively degrade quality of the motion vectors by only decoding a part of the input coded stream to the receiver.

15 "Lossy to lossless" refers to graceful degradation and scalability, implemented in progressive transmission schemes. These deal with situations wherein when transmitting image information over a communication channel, the sender is often not aware of the properties of the output devices such as display size and resolution, and the present requirements of the user, for example when he is browsing through a large image database. To support the large spectrum of image and display sizes and 20 resolutions, the coded bit stream is formatted in such a way that whenever the user or the receiving device interrupts the bit stream, a maximal display quality is achieved for the given bit rate. The progressive transmission paradigm incorporates that the data stream should be interruptible at any stage and still deliver at each breakpoint a good trade-off between reconstruction quality and compression ratio. An interrupted stream 25 will still enable image reconstruction, though not a complete one, which is denoted as a "lossy" approach, since there is loss of information. When the full stream is received a complete reconstruction is possible, hence this is called a "lossless" approach, since no information is lost.

30 Quantization: at the sender or transmitter side of a transmission system, or at any intermediate part or node of the system where quantization is required, a source digital signal S, such as e.g. a source video signal (an image), or more generally any type of input data to be transmitted, is quantized in a quantizer, or in a plurality of quantizers so as to form a number of N bit-streams S₁, S₂, ..., S_N. The source signal can be a function of one or more continuous or discrete variables, and can itself be

continuous or discrete-valued. The generation of bits from a continuous-valued source inevitably involves some form of quantization, which is simply an approximation of a quantity with an element chosen from a discrete set. Each of the generated N bit-streams S_1, S_2, \dots, S_N may or may not be encoded subsequently, for example, entropy encoded, in encoders C_1, C_2, \dots, C_N before transmitting them over a channel.

Quantisation when referred to motion vectors includes setting lengths of motion vector axes (2 for 2D, 3 for 3D) in accordance with an algorithm which chooses between a zero value or a unitary value for each scalar value of the axes of a motion vector. For example, each scalar value of a vector on an axis is compared with a set value, if the scalar value is less than this value, a zero value is assigned for this axis, and if the scalar value is greater than this value a unitary value is assigned.

DETAILED DESCRIPTION OF THE INVENTION

The present invention provides methods and apparatus to compress motion vectors generated by spatial or in-band motion estimation. Spatial or in-band encoders or decoders according to the present invention can be divided into two groups. The first group makes use of algorithms based on motion-vector prediction and prediction-error coding. The second group is based on the integer wavelet transform. The performance of the coding schemes on motion vector sets generated by encoding have been investigated at 3 different sequences at 3 different quality-levels. The experiments show that the encoders/decoders based on motion-vector prediction yield better results than the encoders/decoders based upon the integer wavelet transform. The results indicate that the correlation between the motion vectors seems to degrade as the quality of the decoded images decreases. The encoders/decoders that give the best performance are those based upon either spatio-temporal prediction or spatio-temporal and cross-subband prediction combined with a prediction-error coder. This prediction-error coder codes the prediction errors similarly to the way the DCT coefficients are coded in the JPEG standard for still-image compression.

In a first aspect of the invention the invention discloses an in-band MCTF scheme (IBMCTF), wherein first the overcomplete wavelet decomposition is performed, followed by temporal filtering in the wavelet domain.

A side effect of performing the motion estimation in the wavelet domain is that the number of motion vectors produced is higher than the number of vectors produced by spatial domain motion estimation operating with equivalent parameters. Efficient

compression of these motion vectors is therefore an important issue.

In a second aspect of the invention a number of motion vector coding techniques are presented that are designed to code motion vector data generated by a video codec based on in-band motion estimation and compensation.

5 In an embodiment thereof prediction schemes, using cross subband correlations between motion vectors are exploited.

In an alternative embodiment thereof the use of a table for registration of the most frequently appearing motion vectors for reducing the amount of to code symbols is disclosed.

10 In a further aspect thereof combinations of these motion vector coding techniques is disclosed, in particular the combination of entropy coder 3 with entropy coder 2.

15 The motion vector coding techniques are useful for both the classical "hybrid structure" for video coding, and involves in-band ME/MC as the alternative video codec architecture involving in-band ME/MC and MCTF.

A generic aspect of the motion vector coding techniques is applying a step of classifying the motion vectors before performing a class refining step.

20 In a further aspect of the present invention quality-scalable motion vector coding is used to provide scalable wavelet-based video codecs over a large range of bit-rates. In particular, the present invention includes a motion vector coding technique based on the integer wavelet transform. This scheme allows for reducing the bit-rate spent on the motion vectors. The motion vector field is compressed by performing an integer wavelet transform followed by coding of the transform coefficients using the quad tree coder (e.g. the QT-L coder of P. Schelkens, A. Munteanu, J. Barbarien, M. Galca, X. Giro i Nieto, and J. Cornelis, "Wavelet Coding of Volumetric Medical Datasets," *IEEE Transactions on Medical Imaging, Special issue on "Wavelets in Medical Imaging," Editors M. Unser, A. Aldroubi, and A. Laine*, vol. 22, no. 3, pp. 441-458, March 2003 which is incorporated herewith by reference). In a further aspect of the present invention efficiency of a motion vector coder (MVC) scheme for video processing is improved still further by prediction-based motion vector coder. Embodiments of the present invention combine the compression efficiency of prediction-based MVCs with quality scalability.

30 One aspect of the present invention is a combination of non-linear prediction, e.g. median-based prediction with quality scalable coding of the prediction errors. For

example, the prediction motion vector errors generated by median-based prediction are coded using the QT-L codec mentioned above. However, a drift phenomenon caused by the closed-loop nature of the prediction may result. This means that errors that are successively produced by the quality scalable decoding of the prediction motion vector 5 errors can cascade in such a way that a severely degraded motion vector set is decoded. The following table illustrates this drift effect in a simplified case where the prediction is performed on a 1D dataset for simplicity's sake and each value is predicted by its predecessor. It is preferred to avoid drift.

Original values	1	2	-4	-3	-3	0	4	5	0	1	5	-3
Prediction error (lossless)	1	1	-6	1	0	3	4	1	-5	1	4	-8
Prediction error (lossy)	0	0	-6	0	0	2	4	0	-4	0	4	-8
Decoded values	0	0	-6	-6	-6	-4	0	0	-4	-4	0	-8
Decoding error	-1	-2	-2	-3	-3	-4	-4	-5	-4	-5	-5	-5

10

In a further aspect of the present invention a method and apparatus which includes coding motion information in video processing of a stream of image frames is described for avoiding the drift problem. The method or apparatus is for providing 15 motion vectors of at least one image frame, and for coding the motion vectors to generate a quality-scalable representation of the motion vectors. The quality-scalable representation of motion vectors comprises a set of base-layer motion vectors and a set of one or more enhancement-layers of motion vectors. The method of decoding and a decoder for such coded motion vectors as part of receiving and processing a bit stream 20 at a receiver includes the base-layer of motion vectors being losslessly decoded, while the one or more enhancement layers of motion vectors are progressively received and decoded, optionally including progressive refinement of the motion vectors, eventually up to their lossless reconstruction. This embodiment ensures that the motion vectors are progressively refined at the receiver in a lossy-to-lossless manner as the base-layer 25 of motion vectors is losslessly decoded, while the one or more enhancement layers of motion vectors are progressively received and decoded.

An example of a communication system 210 which can be used with the present invention is shown in Fig. 15. It comprises a source 200 of information, e.g. a source of video signals such as a video camera or retrieval from a memory. The signals are encoded in an encoder 202 resulting in a bit stream, e.g. a serial bit stream 5 which is transmitted through a channel 204, e.g. a cable network, a wireless network, an air interface, a public telephone network, a microwave link, a satellite link. The encoder 202 forms part of a transmitter or transceiver if both transmit and receive functions are provided. The received bit stream is then decoded in a decoder 206 which is part of a receiver or transceiver. The decoding of the signal may provide at 10 least one of spatial scalability, e.g. different resolutions of a video image are supplied to different end user equipments 207 – 209 such as video displays; temporal scalability, e.g. decoded signals with different frame rate/frame number ratios are supplied to different user equipments; and quality scalability, e.g. decoded signals with different signal to noise ratios are supplied to different user equipments.

15 Several motion vector (MV) coding techniques are included within the scope of the present invention to compress motion vector sets. Common generic architectures for motion vector coders and coding methods according to embodiments of the present invention are shown in Figures 1a and b as generally known by the skilled person. The techniques can be classified into at least two basic groups based on 20 whether they use in-band (Figure 1b) or spatial motion vectors (Figure 1a) as their input. In each case frames of framed data such as a sequence of video frames are coded and motion estimation is carried out to obtain motion vectors. These motion vectors are compressed and transmitted with the bit stream. In the decoder the frame data and the motion vectors are decoded and the video reconstructed using the motion 25 vectors in motion compensation of the decoded frame data.

A Video codec based on spatial or in-band motion estimation using the complete-to-overcomplete discrete wavelet transform

30 A first embodiment of the present invention relates to a video codec which follows a classical “hybrid structure” for video coding, and involves, in one aspect, in-band ME/MC. Alternatively, the same techniques may be applied coding of spatial motion vectors.

An alternative video codec architecture involving in-band ME/MC and MCTF is described in Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P.

Schelkens, and J. Cornelis, "Open-loop, in-band, motion-compensated temporal filtering for objective full-scalability in wavelet video coding," ISO/IEC, incorporated by reference. Performing motion estimation directly between corresponding subbands of the wavelet transformed frames produces poor prediction results due to the shift-variance problem. Several solutions for this problem have been suggested in literature G. Van der Auwera, A. Munteanu, P. Schelkens, and J. Cornelis, "Bottom-up motion compensated prediction in the wavelet domain for spatially scalable video coding," *IEEE Electronics Letters*, vol. 38, no. 21, pp. 1251-1253, October 2002, X. Li, L. Kerofski and S. Lei, "All-phase motion compensated prediction in the wavelet domain for high performance video coding," in *Proc. IEEE Int. Conf. Image Processing* (ICIP2001), Thessaloniki, Greece, 2001, vol. 3, pp. 538-541, and F. Verdichio, I. Andreopoulos, A. Munteanu, J. Barbarien, P. Schelkens, J. Cornelis, and A. Pepino, "Scalable video coding with in-band prediction in the complex wavelet transform," *Proceedings of Advanced Concepts for Intelligent Vision Systems* (ACIVS2002), Gent, Belgium, pp. 6, September 9-11, 2002.

A video codec according to an embodiment of the present embodiment is based on the complete-to-overcomplete discrete wavelet transform (CODWT). This transform provides a solution to overcome the shift-variance problem of the discrete wavelet transform (DWT) while still producing critically sampled error-frames is the low-band shift method (LBS) introduced theoretically in H. Sari-Sarraf and D. Brzakovic, "A Shift-Invariant Discrete Wavelet Transform," *IEEE Trans. Signal Proc.*, vol. 45, no. 10, pp. 2621-2626, Oct. 1997 and used for in-band ME/MC in H. W. Park and H. S. Kim, "Motion estimation using Low-Band-Shift method for wavelet-based moving-picture coding," *IEEE Trans. Image Proc.*, vol. 9, no. 4, pp. 577-587, April 2000. Firstly, this algorithm reconstructs spatially each reference frame by performing the inverse DWT. Subsequently, the LBS method is employed to produce the corresponding overcomplete wavelet representation, which is further used to perform in-band ME and MC, since this representation is shift invariant. Basically, the overcomplete wavelet decomposition is produced for each reference frame by performing the "classical" DWT followed by a unit shift of the low-frequency subband of every level and an additional decomposition of the shifted subband. Hence, the LBS method effectively retains separately the even and odd polyphase components of the undecimated wavelet decomposition – see G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Wellesley-Cambridge Press, 1996. The "classical" DWT (i.e. the

critically-sampled transform) can be seen as only a subset of this overcomplete pyramid that corresponds to a zero shift of each produced low-frequency subband, or conversely to the even-polyphase components of each level's undecimated decomposition. An improved form of the complete-to-overcomplete transform is 5 described in US 2003 0133500 which is incorporated herewith in its entirety. This latter document describes a method of digital encoding or decoding a digital bit stream, the bit stream comprising a representation of a sequence of n-dimensional data structures. The method is of the type which derives at least one further subband of an overcomplete representation from a complete subband transform of the data structures, 10 and comprises providing a set of one or more critically subsampled subbands forming a transform of one data structure of the sequence; applying at least one digital filter to at least a part of the set of critically subsampled subbands of the data structure to generate a further set of one or more further subbands of a set of subbands of an overcomplete representation of the data structure, wherein the digital filtering step 15 includes calculating at least a further subband of the overcomplete set of subbands at single rate.

Using the CODWT transform, the overcomplete discrete wavelet transform (ODWT) of a frame can be constructed in a level-by-level manner starting from the critically-sampled wavelet representation of that frame – see G. Van der Auwera, A. 20 Munteanu, P. Schelkens, and J. Cornelis, "Bottom-up motion compensated prediction in the wavelet domain for spatially scalable video coding," *IEE Electronics Letters*, vol. 38, no. 21, pp. 1251-1253, October 2002. The shift-variance problem does not occur when performing motion estimation between the critically-sampled wavelet transform of the current frame and the ODWT of the reference frame, because the 25 ODWT is a shift-invariant transform. The general setup of an in-band video codec based on the CODWT is shown in Figure 1c.

A particular example of this embodiment will now be presented but the motion vector coding techniques of the present invention is not limited thereto. For instance the present invention includes within its scope determining per detail subband motion vectors. In accordance with this example, the in-band motion estimation is performed 30 on a per-level basis. For the highest decomposition level, block-based motion estimation and compensation is performed independently on the LL subband. The motion estimation for the LH, HL and HH subbands is not performed independently. Instead, only one vector is derived for each set of three blocks located at

corresponding positions in the three subbands. This vector minimizes the mean square error (MSE) of the three blocks together. The LH, HL and HH subbands at lower levels can be handled identically. The intra-frames and error-frames are then further encoded. Every frame is predicted with respect to another frame of the video sequence,
5 e.g. a previous frame or the previous frame as the reference, but the present invention is not limited to either selecting a previous frame or a further frame. Also, the block size for the ME/MC is set to 8 pixels, regardless of the decomposition level. The search range is dyadically decreased with each level, starting at [-8, 7] for the first level. Figure 2 exemplifies the motion estimation setup for two decomposition levels.

10

Motion vector coding

The structure of the set of motion vectors produced by the described in-band motion estimation technique for a wavelet decomposition with L levels is shown in Figure 3.

15 Several motion vector (MV) coding techniques are presented to compress motion vector sets of this type all of which are included within the scope of the present invention. The techniques can be classified into at least two groups based on their architecture. The first group of MV coders converts the in-band motion vectors to their equivalent spatial domain vectors and then performs motion vector prediction
20 followed by prediction error coding. A common generic architecture for this group of coders is presented in Figure 4(a). In the following coders and decoders which use in-band coding of the motion vectors will be described but the techniques apply to spatially coded motion vectors as well. As indicated in Figure 4(a) if the input is spatial motion vectors which have been estimated in the spatial domain by spatial
25 motion estimation, then these vectors progress immediately to motion vector prediction and prediction error coding.

In a second type of MV coders, the in-band motion vectors are first converted to their spatial domain equivalents. Afterwards, the components of the equivalent spatial domain vectors are wavelet transformed and the wavelet coefficients are coded.
30 A common architecture for this type of MV coders is shown in Figure 4(b). In the following coders and decoders which use in-band coding of the motion vectors will be described but the techniques apply to spatially coded motion vectors as well. As indicated in Figure 4(b) if the input is spatial motion vectors which have been estimated in the spatial domain by spatial motion estimation, then these vectors go

immediately to the Integer Wavelet transform step followed by coding of the wavelet coefficients.

For all the embodiments of the present invention, where coding is described the present invention also includes decoding by the inverse process to obtain the motion vectors followed by motion compensation of the decoded frame data using the retrieved motion vectors.

For both types of coders, the first step is the conversion of the in-band motion vectors to their equivalent spatial domain motion vectors. The motion vectors generated by in-band motion estimation consist of a pair of numbers (i,j) indicating the horizontal and vertical phase of the ODWT subband where the best match was found, and a pair of numbers (x,y) representing the actual horizontal and vertical offset of the best matching block within the indicated subband. From this data, an equivalent spatial domain motion vector $(x_{\text{spatial}}, y_{\text{spatial}})$ can be derived for each block using the following formulas:

$$\begin{aligned} x_{\text{spatial}} &= ((2 \cdot \text{pel})^{\text{level}} \cdot x + i) \\ y_{\text{spatial}} &= ((2 \cdot \text{pel})^{\text{level}} \cdot y + j) \end{aligned}$$

For more explanation of these formulas see J. Barbarien, I. Andreopoulos, A. Munteanu, P. Schelkens, and J. Cornelis, "Coding of motion vectors produced by wavelet-domain motion estimation," ISO/IEC JTC1/SC29/WG11 (MPEG), Awaji island, Japan, m9249, December 2002. In these formulas, *pel* indicates the accuracy of the motion estimation (*pel* = 1 for integer-pel accuracy, *pel* = 2 for half-pel accuracy and *pel* = 4 for quarter-pel accuracy) and *level* indicates the wavelet decomposition level associated with the in-band motion vector.

The conversion to the equivalent spatial domain vectors is made to simplify the prediction or wavelet transformation that follows it.

The following notations are introduced to facilitate the following description:

- *L*: The number of levels in the wavelet decomposition of the frames.
- $\text{mv}_{\text{tot}}(i)$: The complete set of equivalent spatial domain motion vectors generated by in-band motion estimation between frame *i* and *i*-1.
- $\text{mv}_A(i)$: The set of equivalent spatial domain motion vectors generated by performing motion estimation between the LL subbands of frames *i* and *i*-1. This is a

subset of $mv_{tot}(i)$.

- $mv_D^n(i)$: The set of equivalent spatial domain motion vectors generated by performing motion estimation between the LH, HL and HH subbands of level n of frame i and $i-1$. This is a subset of $mv_{tot}(i)$.
- 5 It is clear that $mv_{tot}(i) = mv_A(i) \cup \left(\bigcup_{k=1}^{k=L} mv_D^k(i) \right)$.

Motion vector coders based on motion-vector prediction and prediction-error coding

An embodiment of an MV coding scheme based on motion vector prediction
10 and prediction error coding will be described with reference to Figure 4(a). Four different motion vector prediction schemes and five different prediction error coders are included as individual embodiments of the present invention. The motion vector prediction schemes will be discussed first.

15 **a) MOTION VECTOR PREDICTION SCHEMES**

Prediction Scheme 1

In scheme 1, the motion vectors in each subset of $mv_{tot}(i)$ are predicted independently of the motion vectors in the other subsets. The prediction of the motion vectors within each subset of $mv_{tot}(i)$ is performed similar to the motion vector prediction in H.263 – see A. Puri and T. Chen, "Multimedia Systems, Standards, and Networks," Marcel Dekker, 2000. Each vector is predicted by taking the median of a number of neighboring vectors. The neighboring vectors that are considered for the default case and for the particular cases that occur at boundaries are shown in Figure 5.

25 *Prediction Scheme 2*

Prediction scheme 1 exploits only the spatial correlations between the neighboring motion vectors within each subset of $mv_{tot}(i)$. The second prediction scheme exploits spatial correlations within the same subset as well as the correlations between corresponding motion vectors in different subsets of $mv_{tot}(i)$. The prediction 30 of a vector in a certain subset is again calculated by taking the median of a set of

vectors. This set consists of a number of spatially neighboring vectors and the vectors at the equivalent position in other subsets of $mv_{tot}(i)$. These other subsets are chosen based upon the wavelet decomposition level corresponding to the predicted vectors' subset. Only subsets corresponding to higher levels are considered. This is done to 5 sustain support for resolution scalability of the motion vector data. The spatially neighboring vectors are chosen in the same way as in scheme 1 (Figure 5). Figure 6 illustrates the prediction scheme in the default case. The boundary cases are handled analogously to scheme 1.

10 *Prediction Scheme 3*

Prediction scheme 3 exploits spatial and temporal correlations between the motion vectors. The prediction of the vectors in $mv_{tot}(i)$ is again performed by calculating the median of a set of vectors. This set consists of spatially neighboring vectors in the same subset of $mv_{tot}(i)$ as the predicted vector, and the vector at the 15 same position as the predicted vector in the motion vector set $mv_{tot}(i-1)$. The prediction algorithm is the same for all subsets since no vectors from other subsets are involved in the prediction. The scheme is illustrated in Figure 7 for the default case. Boundary cases are handled analogously to scheme 1.

Temporal correlations are not exploited for the first set of motion vectors 20 generated at the beginning of a new GOP. For these motion vector sets, scheme 1 is applied.

Prediction Scheme 4

Prediction scheme 4 may be considered as a combination of schemes 2 and 3. 25 Besides spatial correlations, both temporal and cross-subset correlations are exploited. The prediction is again calculated by taking the median of several vectors that are correlated with the predicted vector. In this case, the prediction of a vector in a subset of $mv_{tot}(i)$ involves the spatially neighboring vectors in the same subset, the vector at the same position in the previous motion vector set $mv_{tot}(i-1)$, and the vectors at the 30 corresponding position in subsets associated to higher levels of decomposition. This is illustrated in Figure 8 for the default case. Boundary cases are handled analogously to scheme 1. The prediction scheme processes the first motion vector set in each GOP in

a different way than the other motion vector sets. For the prediction of these particular sets, prediction scheme 2 is used.

b) PREDICTION ERROR CODING

5 Next, the different prediction error coding schemes are discussed. All the presented schemes encode the prediction error components separately. Given the search ranges used in the in-band motion estimation, it can be determined that the components of the prediction error vectors are integer numbers limited to the following intervals:

10

Integer pixel accuracy	[-31,31]
Half-pixel accuracy	[-63,63]
Quarter pixel accuracy	[-127,127]

Table 1: Range of the prediction error components.

This can be verified using the conversion formulas between the in-band motion vectors and their equivalent spatial domain vectors.

15

Prediction-Error Coder 1

This coder uses context-based arithmetic coding to encode the prediction error components. As said before, the x and y components of the prediction error are coded separately. Both components are integer numbers restricted to a bounded interval as specified in Table 1. This interval is divided into several subintervals as specified in the following table (Table 2):

Integer pixel accuracy		Half pixel accuracy		Quarter pixel accuracy	
Interval	Index	Interval	Index	Interval	Index
[-31,-25]	0	[-63,-50]	0	[-127,-111]	0
[-24,-18]	1	[-49,-39]	1	[-110,-94]	1
[-17,-11]	2	[-38,-28]	2	[-93,-77]	2
[-10,-4]	3	[-27,-17]	3	[-76,-60]	3
[-3,3]	4	[-16,-6]	4	[-59,-43]	4
[4,10]	5	[-5,5]	5	[-42,-26]	5

[11,17]	6	[6,16]	6	[-25,-9]	6
[18,24]	7	[17,27]	7	[-8,8]	7
[25,31]	8	[28,38]	8	[9,25]	8
		[39,49]	9	[26,42]	9
		[50,63]	10	[43,59]	10
				[60,76]	11
				[77,93]	12
				[94,110]	13
				[111,127]	14

Table 2: Division of the total range of the prediction error components.

Each error component is coded as an interval-index (symbol), representing the interval it belongs to, followed by the component's offset relative to the lower boundary of that interval. Up to six models are defined for the adaptive arithmetic encoder. For each 5 component x and y , one model is used to code the index of the interval and one model per unique interval size (integer-pel and quarter-pel: one model, half-pel: 2 models) is used to encode the offset relative to the interval's lower boundary.

Prediction-Error Coder 2

10 This coder is similar to coder 1, since it also codes the prediction error components as an index representing the interval it belongs to, followed by the component's offset within the interval. The choice of the intervals and the way the offsets are coded is similar to the way DCT coefficients are coded in the JPEG standard for still-image compression – see W. B. Pennebaker and J. L. Mitchell, *JPEG* 15 still image data compression standard. New York: Van Nostrand Reinhold, 1993.

Table 3 presents the intervals.

Integer pixel accuracy		Half pixel accuracy		Quarter pixel accuracy	
Interval/value	Index	Interval/value	Index	Interval/value	Index
0	0	0	0	0	0
$\{-1\} \cup \{1\}$	1	$\{-1\} \cup \{1\}$	1	$\{-1\} \cup \{1\}$	1
$[-3,-2] \cup [2,3]$	2	$[-3,-2] \cup [2,3]$	2	$[-3,-2] \cup [2,3]$	2
$[-7,-4] \cup [4,7]$	3	$[-7,-4] \cup [4,7]$	3	$[-7,-4] \cup [4,7]$	3
$[-15,-8] \cup [8,15]$	4	$[-15,-8] \cup [8,15]$	4	$[-15,-8] \cup [8,15]$	4

[31,-16] $\cup [16,31]$	5	[31,-16] $\cup [16,31]$	5	[31,-16] $\cup [16,31]$	5
		[-63,-32] $\cup [32,63]$	6	[-63,-32] $\cup [32,63]$	6
				[-127,-64] $\cup [64,127]$	7

Table 3: Division of the total range of the prediction error components in coder 2.

When coding the offset of the prediction error component within the interval, a distinction is made between positive and negative components. For positive components, the value that is coded is equal to the prediction error component. For 5 negative components, the algorithm encodes the sum of the prediction error component and the absolute value of the lower bound of the interval it belongs to. For example, a component value of -12 is coded as symbol 4 (to indicate the interval) followed by 3 ($=-12+|-15|$). It is obvious that no offset is coded for interval 0.

The interval-index and the value for the offset are coded using context-based 10 arithmetic coding. For each component x and y , one model is used to code the interval-index. A different model is used to encode the offset values, and this is done depending on the interval. The offset value is coded differently for the intervals 0 to 4 than for intervals 5 to 7. In the first case the different offset values are directly coded as different symbols of the model. In the second case, the model only allows two 15 symbols 0 and 1, and the offset value is coded in its binary representation.

Prediction-Error Coder 3

Before discussing the different prediction-error coders it has already been mentioned that in principle, the components of the prediction error can only take a 20 limited number of different values. In a usual prediction error set, not all of the possible values occur. The occurrence of very large values is highly unlikely if the employed prediction was effective. This coder accounts for this aspect by transmitting which values do occur in the x and y components of the prediction-error set. It then constructs a lookup table for both components linking a symbol to each of the 25 occurring values and codes the prediction error components based on this lookup tables. Two sequences of bits, one sequence for the x component of the prediction errors and one for the y component indicate the values that occur in the set of

prediction errors. If a value is present in the prediction error set that is going to be coded, the corresponding bit in the sequence is set to 1, otherwise it is set to 0. This is illustrated in Figure 9.

- Referring to Figure 9 a lookup table is constructed for the x and y components,
- 5 linking each value occurring in the prediction error set to a unique symbol. The lookup table is built by numbering the occurring values in a linear way, from the smallest value to the largest one. To encode a prediction error, (1) the corresponding symbols for both components x and y are found in the lookup tables, and (2) the retrieved symbols are entropy coded with an adaptive arithmetic coder that employs different
- 10 models for the x and y components. The conversion to symbols obtained with this algorithm applied on the example shown in Figure 9 is presented in Table 4.

x prediction-error component		y prediction-error component	
Component Value	Symbol	Component value	Symbol
-3	0	-6	0
-2	1	1	1
0	2	7	2
5	3		

Table 4

Prediction-Error Coder 4

- 15 Similar to the motion vectors, the prediction errors can be split into a number of subsets corresponding to different wavelet decomposition levels and/or subbands. Each subset of the prediction errors is coded in the same way. The x and y components of the prediction errors in a subset can be considered as arrays of integer numbers. These arrays are coded using a suitable algorithm such as the quadtree-coding
- 20 algorithm. The quadtree-coding algorithm entropy codes the generated symbols using adaptive arithmetic coding employing different models for the significance, refinement and sign symbols. Such a coder is inherently quality scalable as described in P. Schelkens, A. Munteanu, J. Barbarien, M. Galca, X. Giro i Nieto, and J. Cornelis, "Wavelet Coding of Volumetric Medical Datasets," *IEEE Transactions on Medical Imaging, Special issue on "Wavelets in Medical Imaging," Editors M. Unser, A. Aldroubi, and A. Laine*, vol. 22, no. 3, pp. 441-458, March 2003.
- 25

Prediction-Error Coder 5

In this coding scheme, the prediction error subsets associated to the different wavelet decomposition levels, are arranged in a 3D structure as shown in Figure 10.

- 5 This 3D structure can be split into two three-dimensional arrays of integer numbers by considering the x and y components of the prediction errors separately. These two arrays are then coded using cube splitting algorithm, combined with context-based adaptive arithmetic coding of the generated symbols. Separate sets of models are used for the x and y component arrays. The significance symbols, 10 refinement symbols and sign symbols are entropy coded using separate models.

Motion vector coders based on the integer wavelet transform.*Integer wavelet transform*

- 15 For each subset of $mv_{tot}(i)$, both components of the motion vectors are transformed to the wavelet domain using the (5,3) integer wavelet transform with 2 decomposition levels. The resulting wavelet coefficients are then coded using either quadtree-based coding or cube splitting.

- 20 *Quadtree based wavelet coefficient coding.*

The quadtree based coding is handled in exactly the same way as in prediction error coder 4.

Wavelet coefficient coding using cube splitting

- 25 The cube splitting is handled in exactly the same way as in prediction error coder 5.

The above coders are inherently quality scalable as disclosed in the article by P. Schelkens, A. Munteanu, J. Barbarien, M. Galca, X. Giro i Nieto, and J. Cornelis, mentioned above and incorporated by reference.

- 30

Experimental results

The proposed motion vector coding techniques have been tested on the motion vector sets generated by encoding 3 different sequences at three different quality-

levels. The test sequences are listed in Table 5.

Name	Resolution	Framerate	Number of frames
Football	SIF	30 Hz	100
Mobile	CIF	30 Hz	256
Stefan	CIF	30 Hz	300

Table 5: Overview of the test sequences.

- 5 All encoding runs were done using three wavelet decomposition levels and integer pixel accuracy of the motion estimation. The GOP (Group of picture) size was set to 16 frames.

To calculate the size reductions, the uncompressed size of the motion vector data must first be determined. The structure of the generated motion vector set is 10 shown in Figure 11.

The bits needed to code the ODWT phase components of the in-band motion vectors for the different subsets are listed in Table 6. The amounts of bits needed to represent the offsets within the ODWT subbands are listed in Table 7.

Subset	Horizontal phase i		Vertical phase j	
	Possible values	Bits needed	Possible values	Bits needed
LL subband of level 3	[0, 7]	3 bits	[0, 7]	3 bits
LH,HL and HH subband of level 3	[0, 7]	3 bits	[0, 7]	3 bits
LH,HL and HH subband of level 2	[0, 3]	2 bits	[0, 3]	2 bits
LH,HL and HH subband of level 1	[0,1]	1 bit	[0,1]	1 bit

Table 6: Bits needed to code the in-band motion vector's phase components.

	Horizontal offset x	Vertical offset y

Subset	Possible values	Bits needed	Possible values	Bits needed
LL subband of level 3	[-2,1]	2 bits	[-2,1]	2 bits
LH,HL and HH subband of level 3	[-2,1]	2 bits	[-2,1]	2 bits
LH,HL and HH subband of level 2	[-4,3]	3 bits	[-4,3]	3 bits
LH,HL and HH subband of level 1	[-8,7]	4 bit	[-8,7]	4 bit

Table 7: Bits needed to code the offset components of the in-band motion vectors.

From the two previous tables, it can be derived that the total number of bits needed to represent an in-band motion vector is always equal to 10 irrespective of the subset the motion vector is part of. Together with the information of the structure of the motion

5 vector set (as given in Figure 11), the total uncompressed size of one motion vector set can be calculated. For CIF sequences the number of bits spent per frame equals:

$$(2 \cdot (5 \cdot 4) + (11 \cdot 9) + (22 \cdot 18)) \cdot 10 \text{ bits} = 5350 \text{ bits} = 668.75 \text{ bytes}$$

For SIF sequences the uncompressed size is given by:

$$(2 \cdot (5 \cdot 3) + (11 \cdot 7) + (22 \cdot 15)) \cdot 10 \text{ bits} = 4370 \text{ bits} = 546.25 \text{ bytes}$$

10 The results of the experiments are given in the following tables. The reported numbers are the average size reductions in % obtained with respect to the uncompressed size.

Results for the coders based on motion-vector prediction and prediction-error coding.

15

Technique		Average PSNR of the decoded frames		
Motion vector predictor	Prediction error coder	26,1 dB	29,3 dB	40,3 dB
		% reduction	% reduction	% reduction
1	1	3,7	17,2	28,7

2	1	4,3	14,1	23,5
3	1	6,1	19,2	30,9
4	1	8,0	20,2	31,0
1	2	5,8	21,0	33,5
2	2	5,7	17,3	28,0
3	2	7,7	22,4	34,9
4	2	9,7	23,4	35,3
1	3	3,4	18,1	30,0
2	3	4,2	15,2	25,2
3	3	5,2	19,3	31,3
4	3	7,9	21,0	32,1
1	4	2,1	18,8	32,5
2	4	1,8	15,5	27,6
3	4	3,5	20,1	33,9
4	4	5,1	20,7	33,9
1	5	-4,0	13,6	27,8
2	5	-4,5	9,9	22,5
3	5	-2,4	15,0	29,4
4	5	-0,7	15,8	29,5

Table 8: Results for the "Football" sequence.

Technique		Average PSNR of the decoded frames		
Motion vector predictor	Prediction error coder	26,4 dB	29,6 dB	40,2 dB
		% reduction	% reduction	% reduction
1	1	54,4	62,7	71,2
2	1	50,0	56,3	61,8
3	1	54,8	63,8	73,1
4	1	56,2	63,5	71,0
1	2	58,4	66,4	74,5
2	2	54,8	61,1	66,6
3	2	58,5	67,2	76,2
4	2	59,9	67,0	74,1

1	3	55,1	63,2	71,6
2	3	51,9	58,2	64,0
3	3	55,2	63,9	73,2
4	3	56,9	64,0	71,4
1	4	55,7	64,4	73,4
2	4	50,5	57,3	63,6
3	4	56,2	65,3	75,0
4	4	55,6	63,2	71,0
1	5	53,4	62,3	71,7
2	5	47,9	54,9	61,4
3	5	53,8	63,2	73,3
4	5	53,4	61,2	69,3

Table 9: Results for the "Mobile" sequence.

Technique		Average PSNR of the decoded frames		
Motion vector predictor	Prediction error coder	26,2 dB	29,1 dB	40,0 dB
		% reduction	% reduction	% reduction
1	1	13,4	21,9	32,9
2	1	14,3	20,3	28,4
3	1	14,5	22,9	33,9
4	1	16,9	24,0	33,2
1	2	17,2	26,3	37,4
2	2	17,4	24,3	33,2
3	2	17,2	26,3	37,8
4	2	20,2	27,8	37,5
1	3	14,9	23,7	34,3
2	3	16,0	22,4	30,8
3	3	14,9	23,6	34,8
4	3	18,6	25,8	34,9
1	4	14,4	24,7	36,5
2	4	13,4	21,3	30,6
3	4	14,4	24,6	36,7

4	4	15,9	24,8	35,0
1	5	10,6	21,1	33,3
2	5	9,4	17,4	27,0
3	5	10,5	21,0	33,5
4	5	12,2	21,2	31,8

Table 10: Results for the "Stefan" sequence.

Results for the coders based on the integer wavelet transform.

Technique	Average PSNR of the decoded frames		
Wavelet coefficient coding technique	26,1 dB	29,3 dB	40,3 dB
	% reduction	% reduction	% reduction
Quadtree coding	-5,7	3,9	12,9
Cube splitting	-13,6	-3,2	6,4

Table 11: Results for the "Football" sequence.

Technique	Average PSNR of the decoded frames		
Wavelet coefficient coding technique	26,4 dB	29,6 dB	40,2 dB
	% reduction	% reduction	% reduction
Quadtree coding	31,1	31,1	41,0
Cube splitting	27,4	27,4	37,8

5 Table 12: Results for the "Mobile" sequence.

Technique	Average PSNR of the decoded frames		
Wavelet coefficient coding technique	26,2 dB	29,1 dB	40,0 dB
	% reduction	% reduction	% reduction
Quadtree coding	1,8	9,1	18,8
Cube splitting	-3,3	4,3	14,4

Table 13: Results for the "Stefan" sequence.

Several conclusions can be derived from these results. Firstly, the correlation between the motion vectors seems to decrease as the quality of the decoded frames decreases. The diminished motion estimation effectiveness probably causes the motion vectors to drift further away from the real motion field, which usually consists of highly correlated motion vectors. The second conclusion is that the motion vector coding techniques based on the integer wavelet transform perform worse than any of the techniques based on predictive coding. The best of the prediction-based coders seem to

10

be:

- (1) the algorithm based upon the spatio-temporal prediction scheme (scheme 3) and prediction-error coder 2, and
- (2) the algorithm based on the spatio-temporal-cross-subset prediction scheme 5 (scheme 4) and prediction-error coder 2.

Which of the two predictors performs the best depends on the sequence and on the quality of the decoded frames.

10 **Drift-free prediction-based quality and resolution scalable motion vector coding**

In further embodiments of the present invention the problem of drift is solved by a motion vector coding architecture in accordance with a further embodiment of the present invention . The general setup is shown in Figure 12a which is a coder which can use the flow diagram of Figure 12b.

15 With reference to Figures 12 a and b a spatial or in-band set of motion vectors is obtained by motion estimation. These are quantized to generate a quantized set of motion vectors. If the motion vectors are in-band they are converted to their equivalent motion vectors in the spatial domain as described with reference to Figure 4a. The quantized motion vectors are subjected to motion vector prediction by any of the 20 methods described with reference to Fig. 4a as described above. These quantized motion vectors are then coded in accordance with any of the prediction-based motion vector coding methods described above to form a base layer set of quantized motion vectors. In the receiver the decoding of the base layer follows as described with respect to the embodiments above. One or more new sets of motion vectors are created 25 in accordance with this embodiment to form one or more enhancement layers of motion vectors. This is achieved by generating error vectors by finding the difference between each quantized motion vector and its equivalent input motion vector from which it was derived. These error vectors are then subjected to a progressive compression to form one or more quality scalable enhancement layers. Each error 30 vector is a difference between a motion vector and its quantized equivalent, and each error vector is compressed using a progressive entropy coder. The progressive entropy encoder can be a lossy-to-lossless binary entropy encoder. The base layer set and the set or sets of the one or more enhancement layer coded motion vectors are then combined to form the bit stream to be transmitted. Decoding follows by the reverse

procedure.

In accordance with an embodiment of the present invention, the quantization of the input motion vector set can be performed, e.g. by dropping the information on the lowest bit-plane(s). The quantized motion vectors are thereafter compressed using a prediction-based motion vector coding technique, e.g. one of the techniques described in J. Barbarien, I. Andreopoulos, A. Munteanu, P. Schelkens, and J. Cornelis, "Coding of motion vectors produced by wavelet-domain motion estimation," ISO/IEC JTC1/SC29/WG11 (MPEG), Awaji island, Japan, m9249, December 2002 or any of the prediction-based motion vector coding technique described above with respect to the previous embodiments. The resulting compressed data forms the base-layer of the final bit-stream. To avoid drift, this base-layer is preferably always decoded losslessly. Then the quantization error (the difference between the quantized motion vectors and the original motion vectors) is coded in a bit-plane-by-bit-plane manner using a binary entropy coder or a bit-plane coding algorithm supporting quality scalability, e.g. EBCOT described in D. Taubman and M. W. Marcellin, "JPEG2000 - Image Compression: Fundamentals, Standards and Practice," Hingham, MA: Kluwer Academic Publishers, 2001, or QT-L described in P. Schelkens, A. Munteanu, J. Barbarien, M. Galca, X. Giro i Nieto, and J. Cornelis, "Wavelet Coding of Volumetric Medical Datasets," *IEEE Transactions on Medical Imaging, Special issue on Wavelets in Medical Imaging*, Editors M. Unser, A. Aldroubi, and A. Laine, vol. 22, no. 3, pp. 441-458, March 2003. The compressed data forms the enhancement layer(s) of the final bit-stream. The quality and bit-rate of this layer can be varied without introducing drift. In this way, the final bit-stream supports fine-grain quality scalability with a bit-rate that can vary between the bit-rate needed to code the base-layer losslessly and the bit-rate needed for a completely lossless reconstruction of the motion vectors. The bit-rate needed to code the base-layer can be controlled in the encoder by choosing an appropriate quantizer. Choosing a lower bit-rate for the base-layer will however decrease the overall coding efficiency of the entire scheme.

30 Implementation

Fig. 14 shows the implementation of a coder/decoder which can be used with any of the embodiments of the present invention implemented using a microprocessor 230 such as a Pentium IV from Intel Corp. USA. The microprocessor 230 may have an optional element such as a co-processor 224, e.g. for arithmetic operations or

microprocessor 230-224 may be a bit-sliced processor. A RAM memory 222 may be provided, e.g. DRAM. Various I/O (input/output) interfaces 225, 226, 227 may be provided, e.g. UART, USB, I²C bus interface as well as an I/O selector 228. FIFO buffers 232 may be used to decouple the processor 230 from data transfer through these interfaces. A keyboard and mouse interface 234 will usually be provided as well as a visual display unit interface 236. Access to an external memory such as a disk drive may be provided via an external bus interface 238 with address, data and control busses. The various blocks of the circuit are linked by suitable busses 231. The interface to the channel is provided by block 242 which can handle the encoded video frames as well as transmitting to and receiving from the channel. Encoded data received by block 242 is passed to the processor 230 for processing.

Alternatively, this circuit may be constructed as a VLSI chip around an embedded microprocessor 230 such as an ARM7TDMI core designed by ARM Ltd., UK which may be synthesized onto a single chip with the other components shown. A zero wait state SRAM memory 222 may be provided on-chip as well as a cache memory 224. Various I/O (input/output) interfaces 225, 226, 227 may be provided, e.g. UART, USB, I²C bus interface as well as an I/O selector 228. FIFO buffers 232 may be used to decouple the processor 230 from data transfer through these interfaces. A counter/timer block 234 may be provided as well as an interrupt controller 236. Access to an external memory may be provided an external bus interface 238 with address, data and control busses. The various blocks of the circuit are linked by suitable busses 231. The interface to the channel is provided by block 242 which can handle the encoded video frames as well as transmitting to and receiving from the channel. Encoded data received by block 242 is passed to the processor 230 for processing.

Software programs may be stored in an internal ROM (read only memory) 246 which may include software programs for carrying out decoding and/or encoding in accordance with any of the methods of the present invention including motion vector coding or decoding in accordance with any of the methods of the present invention. The methods described above may be written as computer programs in a suitable computer language such as C and then compiled for the specific processor in the design. For example, for the embedded ARM core VLSI described above the software may be written in C and then compiled using the ARM C compiler and the ARM assembler. Reference is made to "ARM System-on-chip", S. Furber, Addison-Wiley,

2000. The present invention also includes a data carrier on which is stored executable code segments, which when executed on a processor such as 230 will execute any of the methods of the present invention, in particular will execute any of the motion vector coding or decoding methods of the present invention. The data carrier may be
5 any suitable data carrier such as diskettes ("floopy disks"), optical storage media such as CD-ROMs, DVD ROM's, tape drives, hard drives, etc. which are computer readable.

Fig. 15 shows the implementation of a coder/decoder which can be used with the present invention implemented using a dedicated motion vector coding module.
10 Reference numbers in Fig. 15 which are the same as the reference numbers in Fig. 14 refer to the same components – both in the microprocessor and the embedded core embodiments.

Only the major differences of Fig. 15 will be described with respect to Fig. 14. Instead of the microprocessor 230 carrying out methods required to provide motion
15 vector compression of a bitstream this work is now taken over by a module 240. Module 240 may be constructed as an accelerator card for insertion in a personal computer. The module 240 has means for carrying out motion vector decoding and/or encoding in accordance with any of the methods of the present invention. These motion vector coding means may be implemented as a separate module 241, e.g. an
20 ASIC (Application Specific Integrated Circuit) or an FPGA (Field Programmable Gate Array) having means for motion vector compression according to any of the embodiments of the present invention described above.

Similarly, if an embedded core is used such as an ARM processor core or an FPGA, a module 240 may be used which may be constructed as a separate module in a
25 multi-chip module (MCM), for example or combined with the other elements of the circuit on a VLSI. The module 240 has means for carrying out motion vector decoding and/or encoding in accordance with any of the methods of the present invention. As above, these means for motion vector coding or decoding may be implemented as a separate module 241, e.g. an ASIC (Application Specific Integrated Circuit) or an
30 FPGA (Field Programmable Gate Array) having means for motion vector encoding or decoding according to any of the embodiments of the present invention described above. The present invention also includes other integrated circuits such as ASIC's or FPGA's which carry out such functions.

Claims

1. Method of coding motion information in video processing of a stream of image frames, comprising:
 - 5 providing motion vectors for at least one image frame,
quantizing the motion vectors to generate a set of quantized motion vectors equivalent to the motion vectors,
compressing the quantized motion vectors losslessly,
generating error vectors, each error vector being a difference between a motion
10 vector and its quantized equivalent, and
progressively encoding the error vectors in a lossy-to-lossless manner.
 2. The method of claim 1 wherein the coding is drift free.
 3. The method of claim 1 or 2, wherein the compression is prediction based.
 4. The method according to any previous claim wherein the motion vectors are
15 resolution scalable.
 5. The method of any previous claim wherein the motion vectors are in-band motion
vectors.
 6. The method according to any previous claim wherein motion vector quality is
scalable.
 - 20 7. The method according to any previous claim wherein the coding is temporally
scalable.
 8. The method according to any previous claim wherein each error vector is a
difference between a motion vector and its quantized equivalent, and each error
vector is compressed using a progressive entropy coder.
 - 25 9. The method according to claim 8, wherein the progressive entropy encoder is a
lossy-to-lossless binary entropy encoder.
 10. The method according to any previous claim, wherein the compression of the
quantized motion vectors is based on motion-vector prediction and prediction-error
coding.
 - 30 11. The method according to claim 10, wherein the prediction error vectors are the
difference between the quantized motion vectors and their predicted equivalent.
 12. The method according to any previous claim wherein the prediction of the
quantized motion vectors is non-linear.
 13. The method according to claim 12, wherein the non-linear prediction includes

- taking the median.
14. The method according to any previous claim, wherein the compression of the prediction-error vectors is done by reducing the alphabet prior to entropy coding.
 15. The method according to any of the claims 1 to 13, wherein the compression of the prediction-error vectors is done by prior classification.
5
 16. A method of decoding encoded motion vectors in a bitstream received at a receiver having been encoded by any of the methods of claims 1 to 15, the method comprising progressively decoding the error vectors in a lossy-to-lossless manner.
 17. The method according to claim 16, further comprising determining quantized motion vectors from received data in the bitstream and reconstructing motion vectors from the quantized motion vectors and the decoded error vectors.
10
 18. The method of claim 17, further comprising predicting the quantized motion vectors from received data in the bitstream.
 19. The method according to claim 17 or 18, further comprising motion compensating decoded frame data retrieved from the bitstream using the reconstructed motion vectors.
15
 20. A method of providing a representation of motion information in video processing of a stream of image frames, comprising:
providing in-band motion vectors of at least one image frame,
20
converting the in-band motion vectors to a spatial domain to generate motion vectors equivalent to the in-band motion vectors,
non-linearly predicting prediction motion vectors from spatial correlation of neighbouring motion vectors in one image frame,
generating prediction-error vectors from differences between the motion vectors in
25
the spatial domain and the prediction motion vectors,
coding the prediction error vectors, and
outputting the coded prediction-error vectors.
 21. The method according to claim 20, wherein the motion vectors are resolution scalable.
 - 30
22. The method according to claim 20 or 21 wherein the coding is temporally scalable.
 23. The method of any of the claims 20 to 22 wherein the coding is drift free.
 24. A method of decoding encoded motion vectors in a bitstream received at a receiver having been encoded by any of the methods of claims 20 to 23, the method comprising progressively decoding the coded prediction error vectors.

25. The method according to claim 24, further comprising predicting motion vectors from data received in the bit stream and reconstructing motion vectors from the predicted motion vectors and the decoded prediction error vectors.
26. A method of providing a representation of motion information in video processing of a stream of image frames, comprising:
 - 5 providing in-band motion vectors of at least one image frame,
 - converting the in-band motion vectors to a spatial domain to generate motion vectors equivalent to the in-band motion vectors,
 - transforming the motion vectors in the spatial domain to a wavelet domain using
 - 10 an integer wavelet transform to generate wavelet coefficients, and
 - coding the wavelet coefficients.
27. The method of claim 26, wherein the coding of the wavelet coefficients is done using 2D or 3D techniques preferably based on quadtree coding or cube splitting respectively.
- 15 28. The method according to claim 26 or 27, wherein the resolution is scalable.
29. The method according to any of the claims 26 to 28, wherein the coding is temporally scalable.
30. The method according to any of the claims 26 to 29, wherein the motion vectors quality scalable.
- 20 31. A method of decoding a bitstream received at a receiver which has been coded by a method according to any of the claims 26 to 29, the method comprising decoding the wavelet coefficients and generating the motion vectors.
32. A method of coding motion vectors of at least one image frame in video processing of a stream of image frames, comprising:
 - 25 transforming the motion vectors using the integer wavelet transform to generate wavelet coefficients, and
 - coding the wavelet coefficients.
33. The method according to claim 32, wherein the motion vectors are in-band.
34. The method according to claim 32, further comprising converting of the in-band motion vectors to their spatial-domain equivalents.
- 30 35. The method according to any of the claims 32 to 34, further comprising transforming the motion vectors using the integer wavelet transform to generate wavelet coefficients.
36. The method according to any of the claims 32 to 35, further comprising coding of

- the wavelet coefficients using 2D or 3D techniques preferably based on quadtree coding or cube splitting respectively.
37. The method according to any of the claims 32 to 36, wherein the resolution is scalable.
- 5 38. The method according to any of the claims 32 to 37, wherein the coding is temporally scalable.
39. The method according to any of the claims 32 to 38, wherein the motion vectors are quality scalable.
40. A method of decoding a bitstream received at a receiver which has been coded by
- 10 a method according to any of the claims 32 to 39, the method comprising decoding the wavelet coefficients and generating motion vectors from the decoded wavelet coefficients.
41. A method of coding motion information in video processing of a stream of image frames, comprising:
- 15 providing motion vectors of at least one image frame, and coding of the motion vectors to generate a quality-scalable representation of the motion vectors.
42. The method according to claim 41, further comprising non-linear motion vector prediction followed by quad-tree coding of the prediction errors.
- 20 43. The method of claim 41 or 42, further comprising a drift-free quality-scalable coding technique for the motion-vectors from a spatial-domain motion estimation obtained by using an integer wavelet transform followed by applying quadtree or cube-splitting coding of the resulting wavelet coefficients.
44. The method according to any of the claims 41 to 43, wherein the resolution is
- 25 scalable.
45. The method according to any of the claims 41 to 44, wherein the coding is temporally scalable.
46. The method according to any of the claims 41 to 45 wherein the coding is drift-free.
- 30 47. The method of any of the claims 41 to 47, the quality-scalable representation of motion vectors comprises a base-layer set of motion vectors and a set of motion vectors in one or more enhancement-layers.
48. A method of decoding a bitstream received at a receiver which has been coded by a method according to claim 47, the method comprising decoding a base layer of

motion vectors and an enhancement layer of motion vectors and enhancing a quality of a decoded image by improving the quality of the base layer of motion vectors using the enhancement layer of motion vectors.

49. An encoder for coding motion information in video processing of a stream of image frames, comprising:
 - means for providing motion vectors for at least one image frame,
 - means for quantizing the motion vectors to generate a set of quantized motion vectors equivalent to the motion vectors,
 - means for compressing the quantized motion vectors losslessly,
- 10 means for generating error vectors, each error vector being a difference between a motion vector and its quantized equivalent, and
- means for progressively encoding the error vectors in a lossy-to-lossless manner.
50. The encoder of claim 49 wherein the coding is drift free.
51. The encoder of claim 49 or 50, wherein the means for compression includes means for prediction based compression.
- 15 52. The encoder according to any of the claims 49 to 51, wherein the means for generating error vector determines a difference between a motion vector and its quantized equivalent, further comprising a progressive entropy coder for compressing each error vector.
- 20 53. The encoder according to claim 52, wherein the progressive entropy encoder is a lossy-to-lossless binary entropy encoder.
54. The encoder according to any of the claims 49 to 53, wherein the means for compression of the quantized motion vectors includes means for motion-vector prediction and prediction-error coding.
- 25 55. The encoder according to claim 54, wherein means for prediction error coding determines error vectors from the difference between the quantized motion vectors and their predicted equivalent.
56. The encoder according to any of the claims 49 to 55, wherein the means for prediction of the quantized motion vectors is a non-linear prediction means.
- 30 57. The encoder according to any of the claims 49 to 56, wherein the means for compression of the prediction-error vectors includes means for reducing the alphabet prior to entropy coding.
58. The method according to any of the claims 49 to 56, wherein the means for compression of the prediction-error vectors includes means for prior classification.

59. A decoder for decoding encoded motion vectors in a bitstream received at the decoder having been encoded by any of the methods of claims 1 to 15, the decoder comprising means for progressively decoding the error vectors in a lossy-to-lossless manner.
- 5 60. The decoder according to claim 59, further comprising means for determining quantized motion vectors from received data in the bitstream and means for reconstructing motion vectors from the quantized motion vectors and the decoded error vectors.
- 10 61. The decoder of claim 60, further comprising means for predicting the quantized motion vectors from received data in the bitstream.
62. The decoder according to claim 60 or 61, further comprising means for motion compensating decoded frame data retrieved from the bitstream using the reconstructed motion vectors.
- 15 63. A device for providing a representation of motion information in video processing of a stream of image frames, comprising:
 - means for providing in-band motion vectors of at least one image frame,
 - means for converting the in-band motion vectors to a spatial domain to generate motion vectors equivalent to the in-band motion vectors,
 - means for non-linearly predicting prediction motion vectors from spatial correlation of neighbouring motion vectors in one image frame,
 - means for generating prediction-error vectors from differences between the motion vectors in the spatial domain and the prediction motion vectors,
 - means for coding the prediction error vectors, and
 - means for outputting the coded prediction-error vectors.
- 20 64. A decoder for decoding encoded motion vectors in a bitstream received at the decoder having been encoded by any of the methods of claims 20 to 23, the decoder comprising means for progressively decoding the coded prediction error vectors.
- 25 65. The decoder according to claim 64, further comprising means for predicting motion vectors from data received in the bit stream and means for reconstructing motion vectors from the predicted motion vectors and the decoded prediction error vectors.
- 30 66. A device for providing a representation of motion information in video processing of a stream of image frames, comprising:

- means for providing in-band motion vectors of at least one image frame,
means for converting the in-band motion vectors to a spatial domain to generate
motion vectors equivalent to the in-band motion vectors,
means for transforming the motion vectors in the spatial domain to a wavelet
domain using an integer wavelet transform to generate wavelet coefficients, and
means for coding the wavelet coefficients.
- 5 67. The device of claim 66, wherein the means for coding of the wavelet coefficients
includes means for quadtree coding or cube splitting.
68. A decoder for decoding a bitstream received at the decoder which has been coded
10 by a method according to any of the claims 26 to 29, the decoder comprising
means for decoding the wavelet coefficients and means for generating the motion
vectors.
69. An encoder for coding motion vectors of at least one image frame in video
processing of a stream of image frames, comprising:
15 means for transforming the motion vectors using the integer wavelet transform to
generate wavelet coefficients, and
means for coding the wavelet coefficients.
70. The encoder according to claim 69, wherein the motion vectors are in-band, further
comprising means for converting the in-band motion vectors to their spatial-
20 domain equivalents.
71. The encoder according to claims 69 or 70, further comprising means for
transforming the motion vectors using the integer wavelet transform to generate
wavelet coefficients.
72. The encoder according to any of the claims 69 to 71, further comprising means for
25 coding of the wavelet coefficients using 2D or 3D techniques preferably based on
quadtree coding or cube splitting respectively.
73. A decoder for decoding a bitstream received at the decoder which has been coded
by a method according to any of the claims 32 to 39, the decoder comprising
means for decoding the wavelet coefficients and means for generating the motion
30 vectors from the decoded wavelet coefficients.
74. An encoder for coding motion information in video processing of a stream of
image frames, comprising:
means for providing motion vectors of at least one image frame, and
means for coding of the motion vectors to generate a quality-scalable

- representation of the motion vectors.
75. The encoder according to claim 74, further comprising means for non-linear motion vector prediction followed by quad-tree coding of the prediction errors.
76. The encoder of claim 74 or 75, further comprising means for applying an integer wavelet transform followed by applying quadtree or cube-splitting coding of the resulting wavelet coefficients.
77. The encoder of any of the claims 74 to 76, further comprising means for generating the quality-scalable representation of motion vectors comprises means for generating a base-layer set of motion vectors and a set of motion vectors in one or more enhancement-layers.
78. A decoder for decoding a bitstream received at a receiver which has been coded by a method according to claim 47, the decoder comprising means for decoding a base layer of motion vectors and an enhancement layer of motion vectors and means for enhancing a quality of a decoded image by improving the quality of the base layer of motion vectors using the enhancement layer of motion vectors.
79. A computer program product which when executed on a processing device executes any of the methods of claims 1 to 48.
80. A machine readable data carrier storing the computer program product according to claim 79.
81. A computer program product which when executed on a processing device implements any of the coders of claims 49 to 78.
82. A machine readable data carrier storing the computer program product according to claim 81.

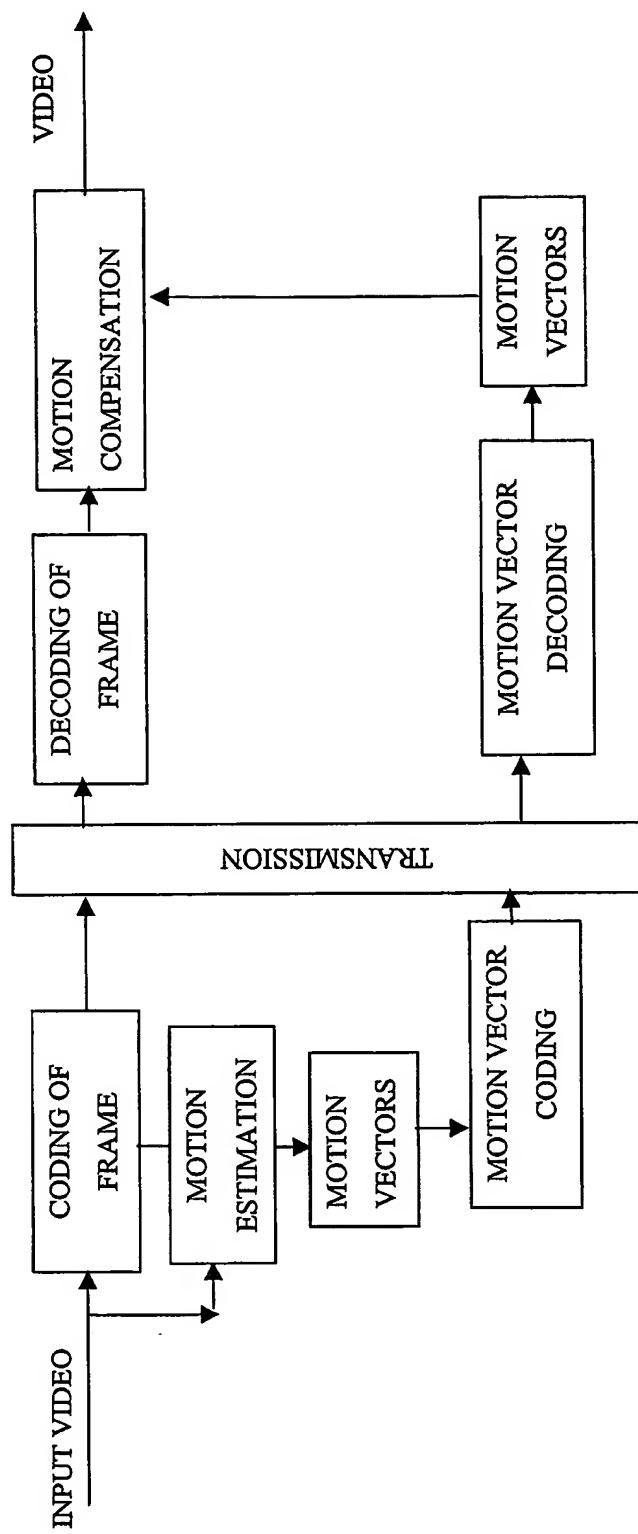


Figure 1(a)

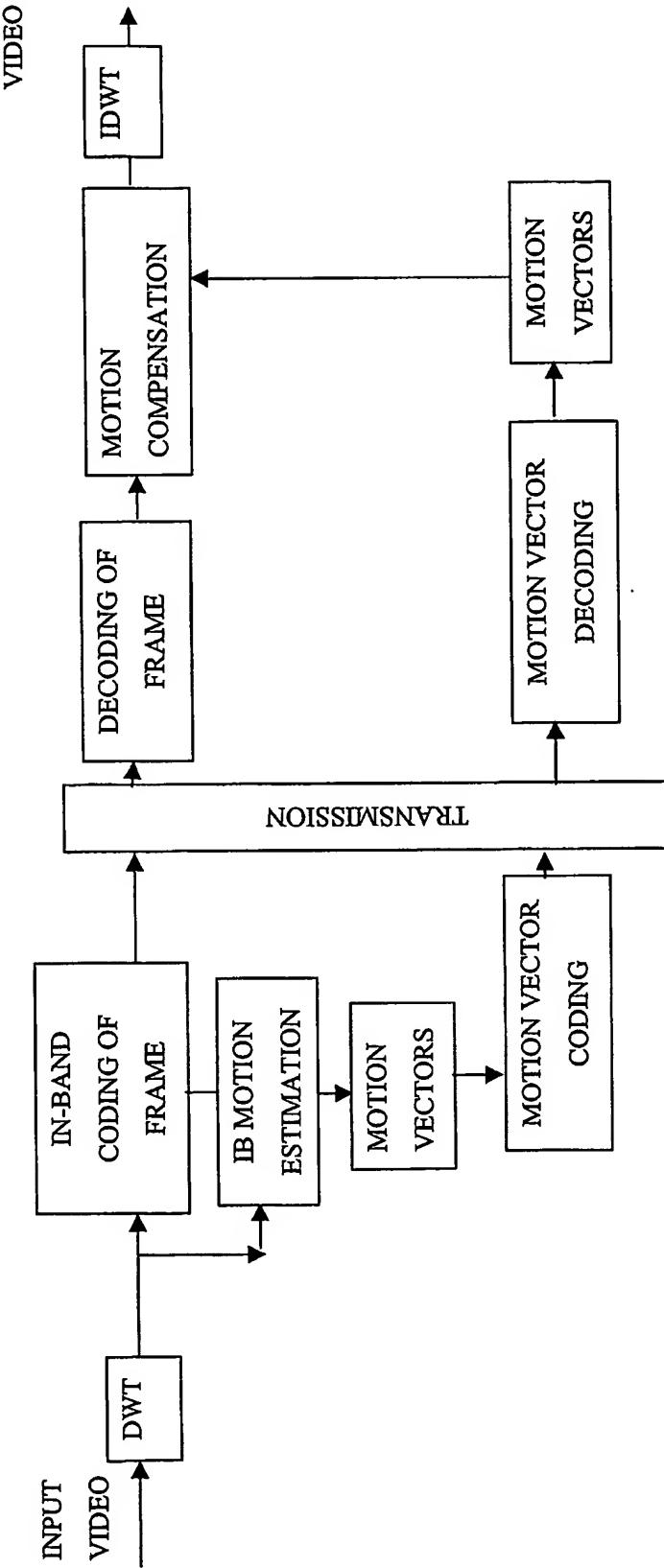


Figure 1(b)

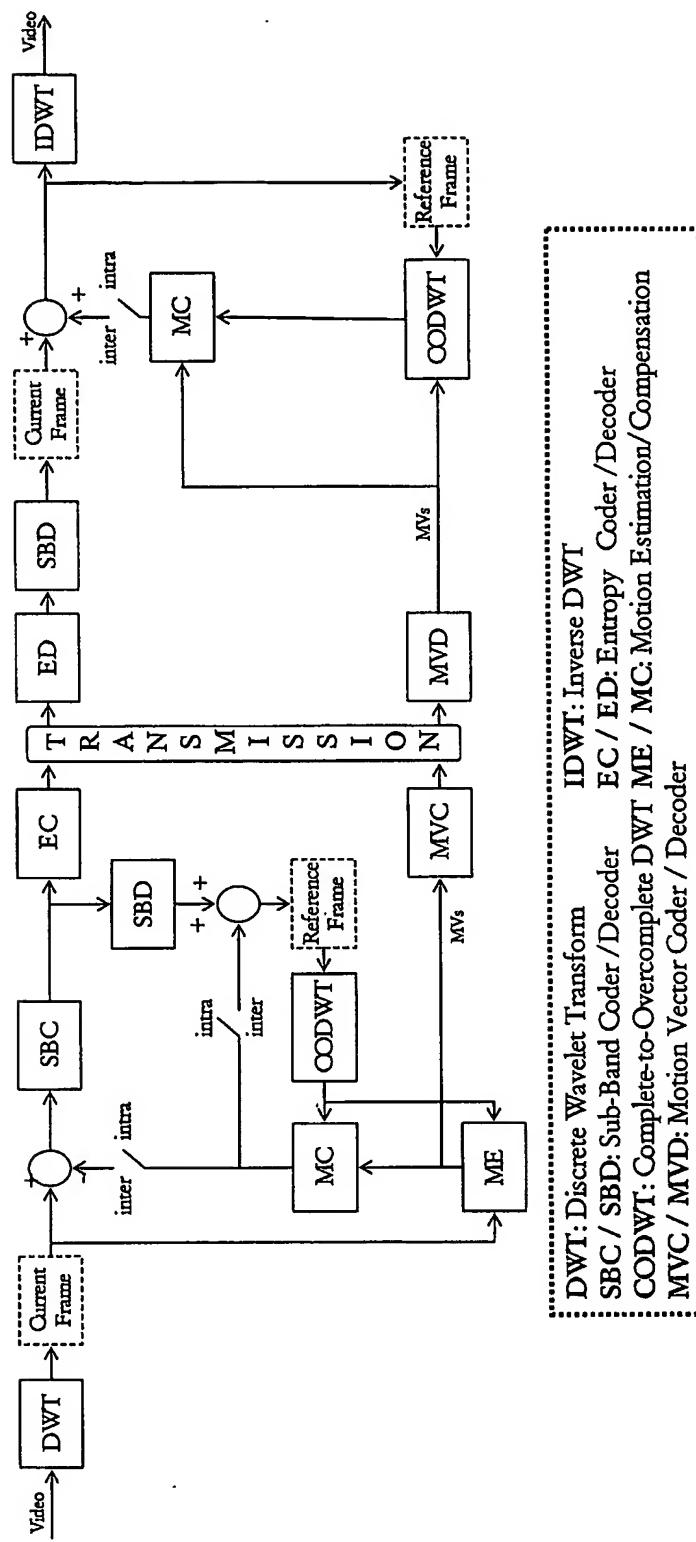


Figure 1 (c)

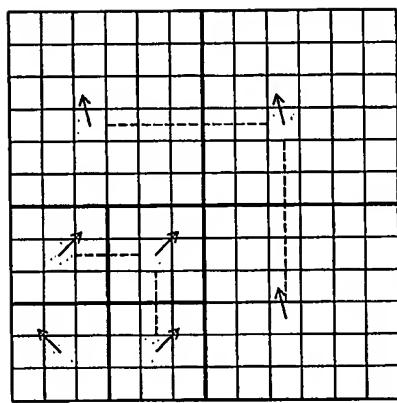


Figure 2

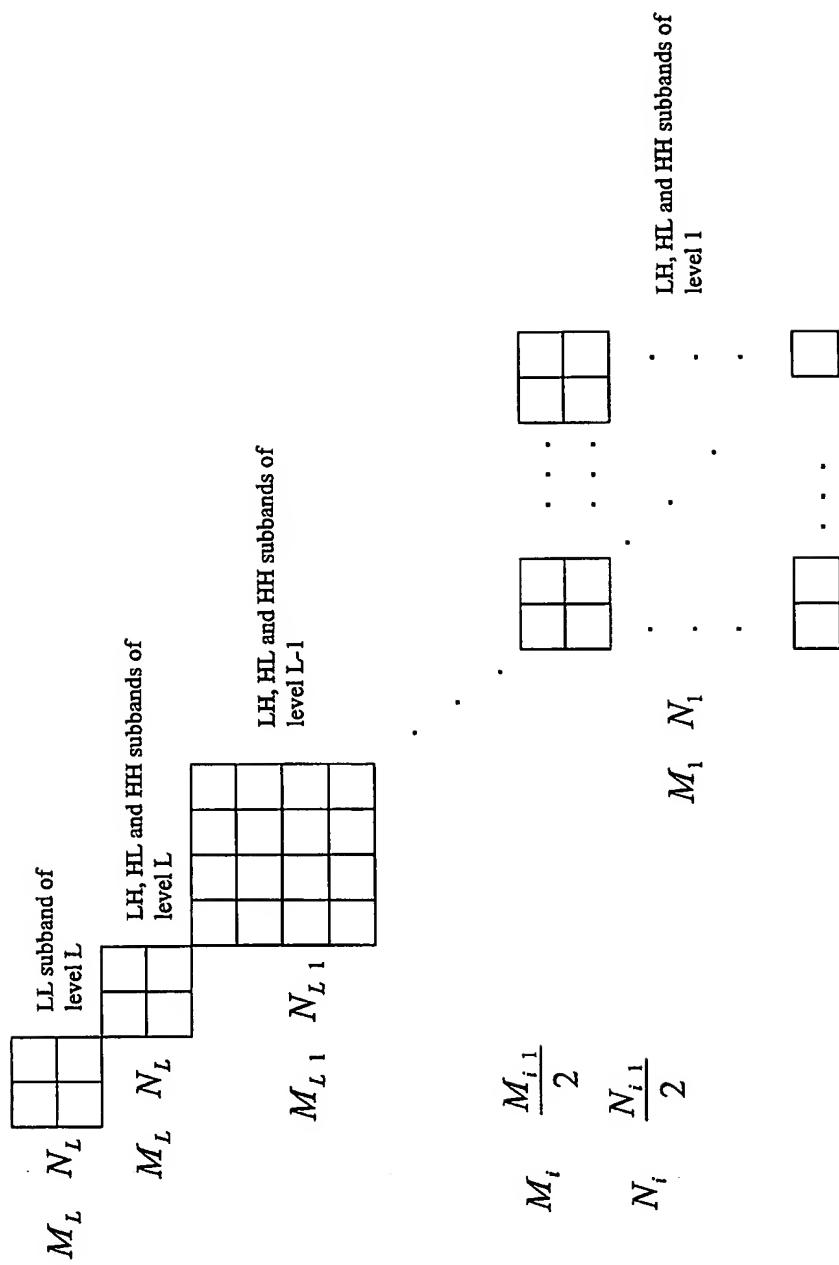


Figure 3

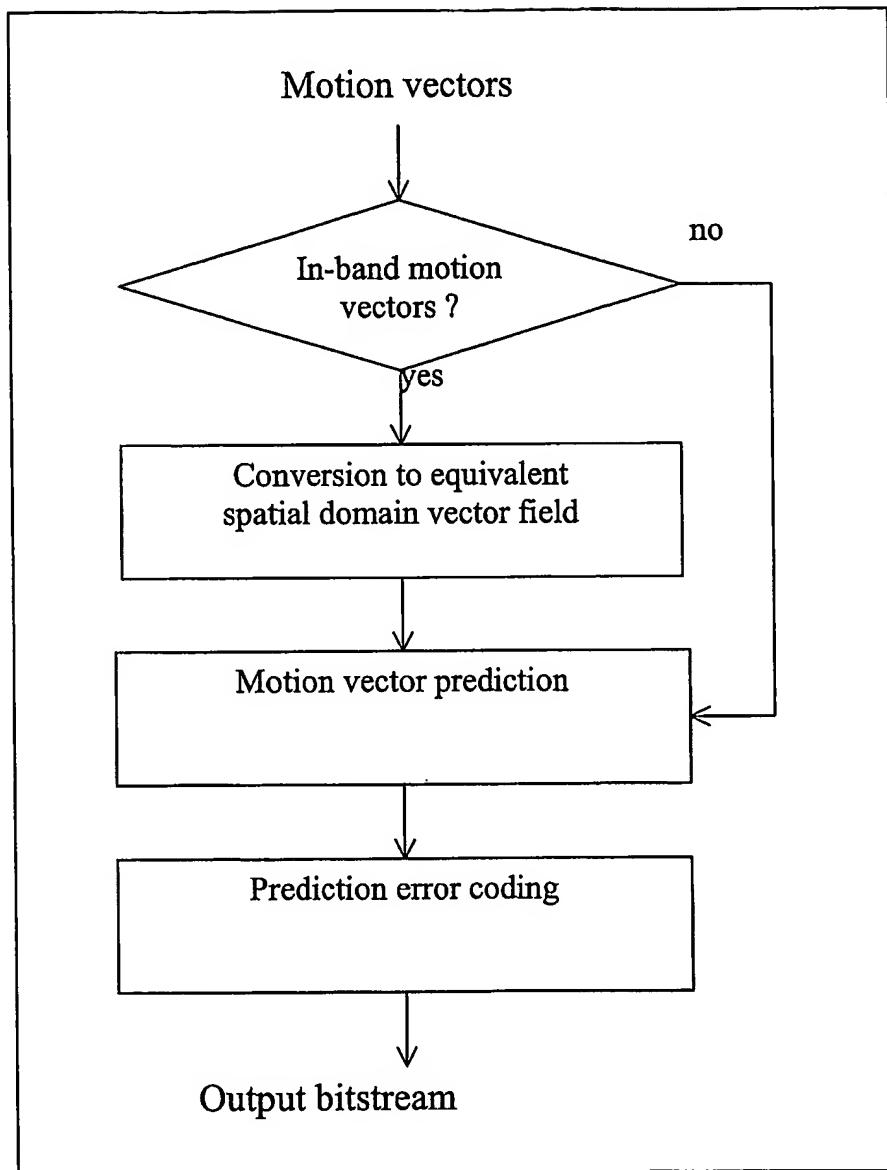


Figure 4(a)

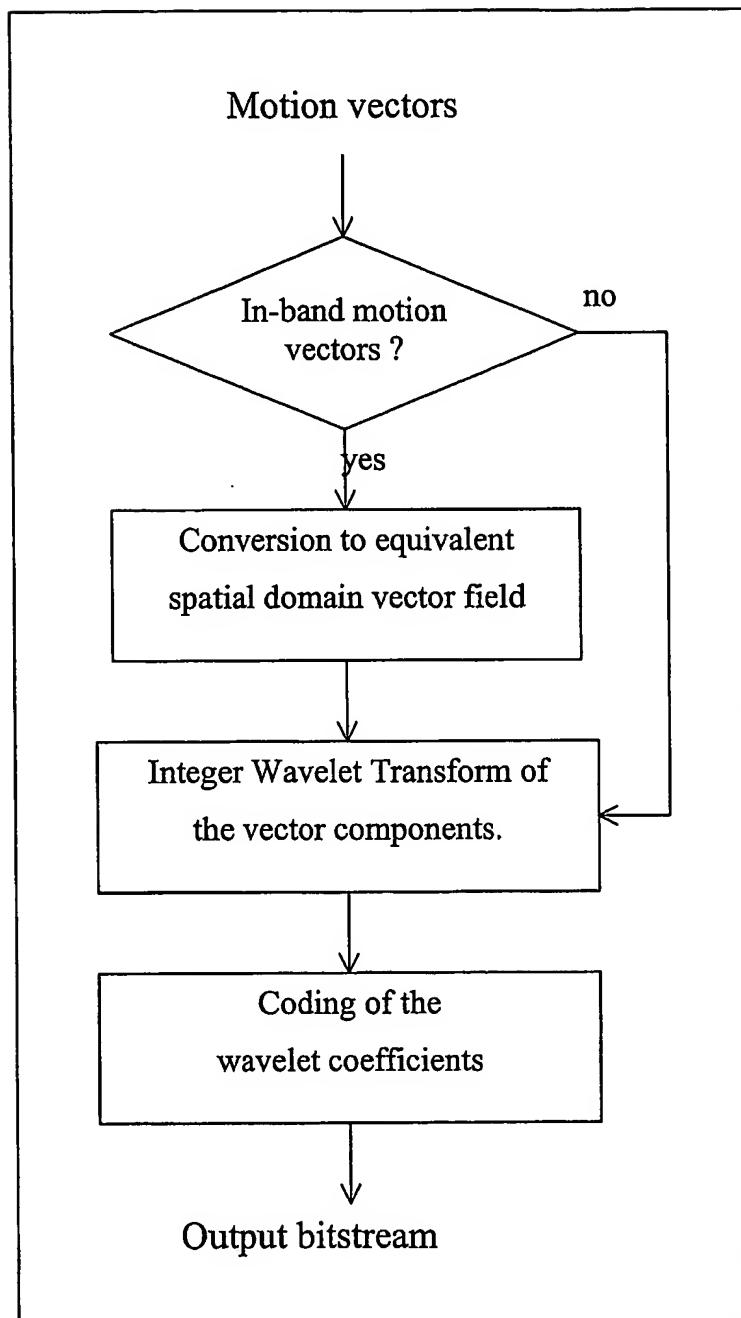


Figure 4 (b)

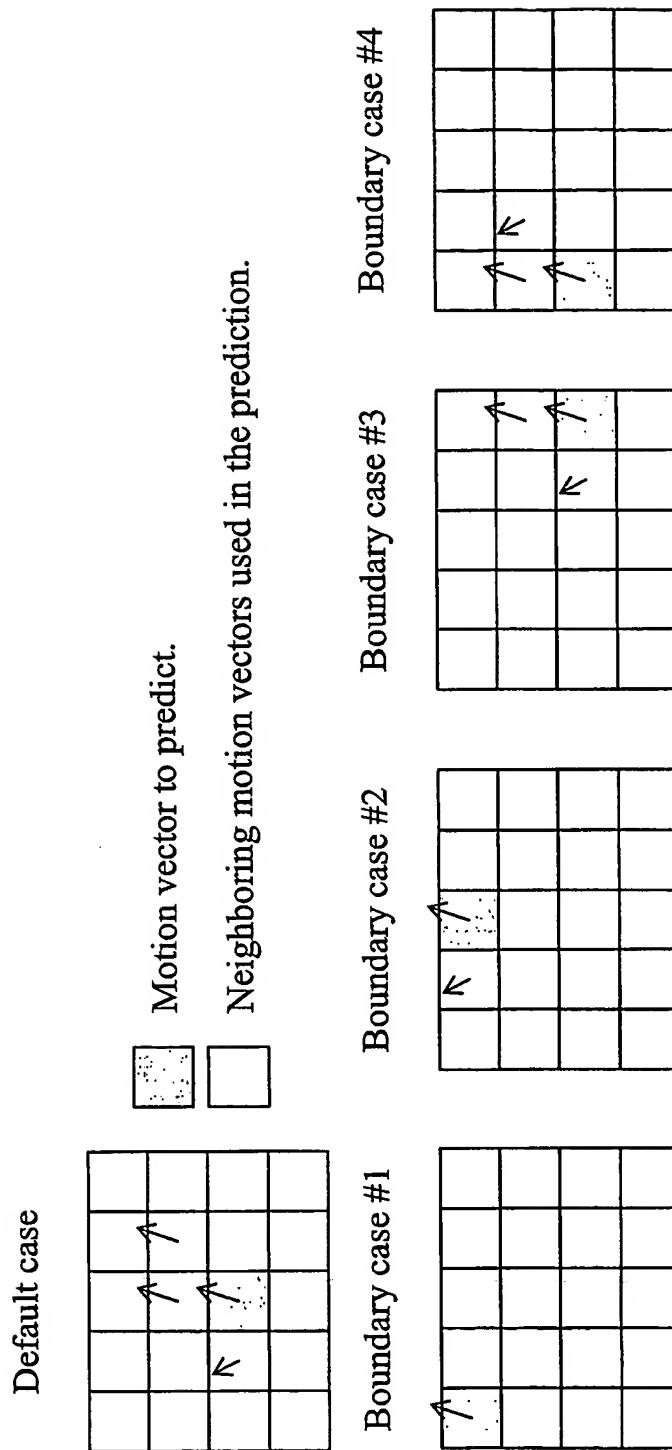
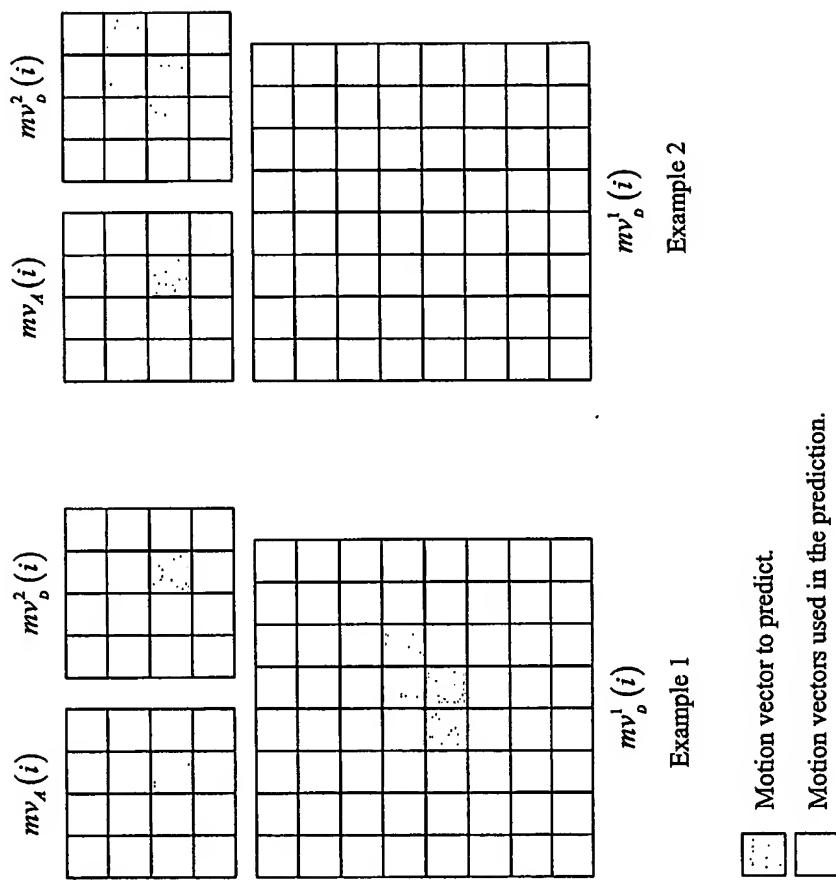


Figure 5

**Figure 6**

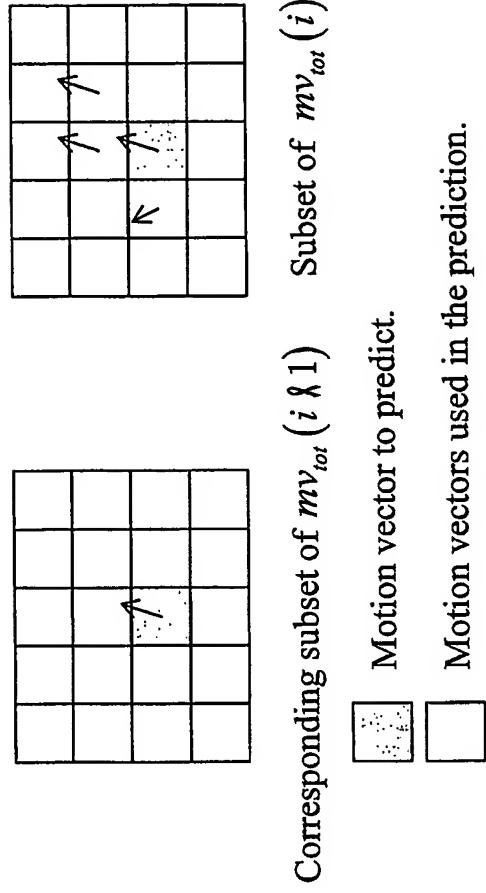


Figure 7

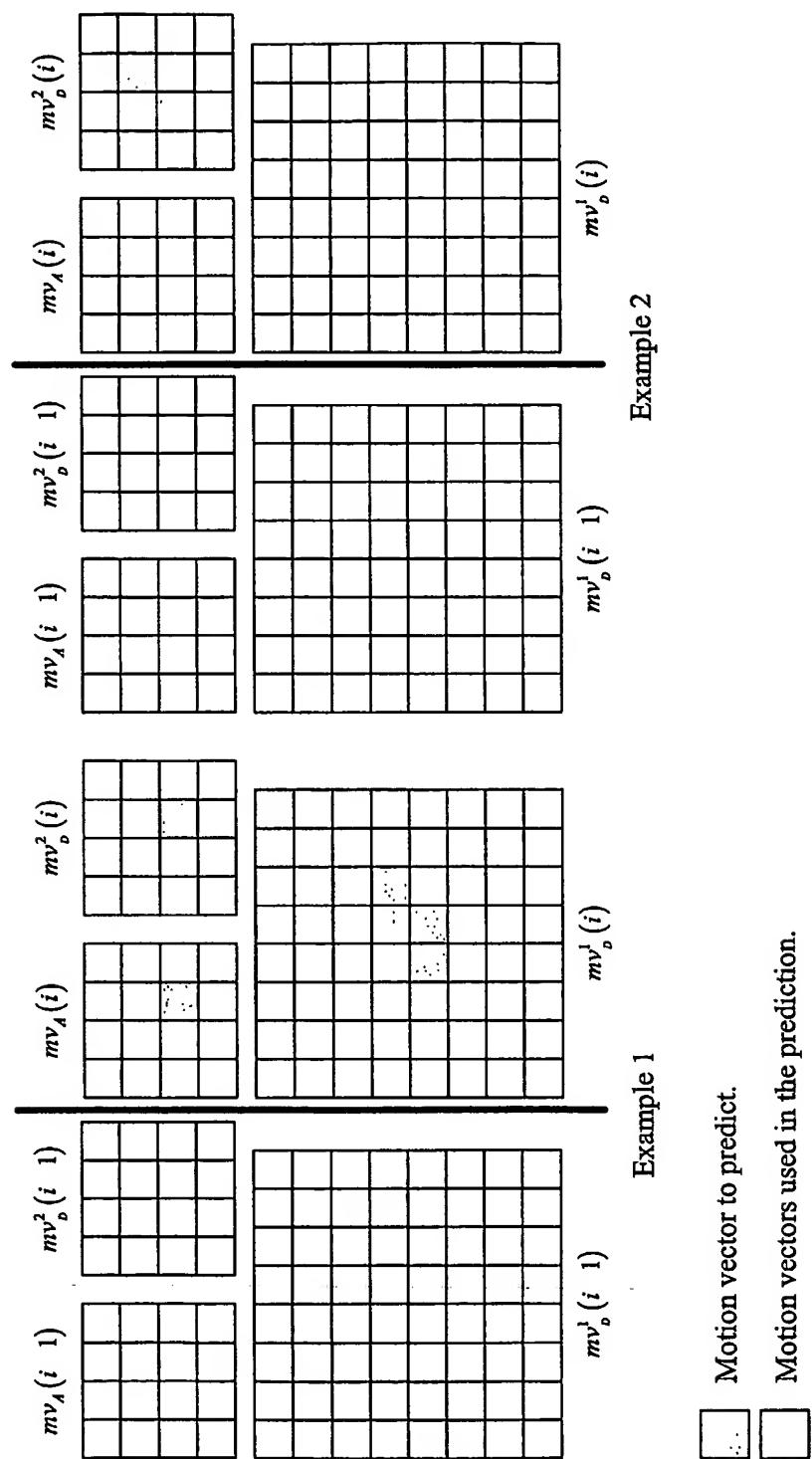


Figure 8

- Motion vector to predict.
- Motion vectors used in the prediction.

Prediction errors (integer-pel ME)

(-3,1)	(5,-6)
(0,1)	(-2,7)

Set of flags for x component:

00
 | / \ |
 31 5 0 -2 -3 -31

Set of flags for y component:

00
 | | |
 31 7 1 -6 -31

Figure 9

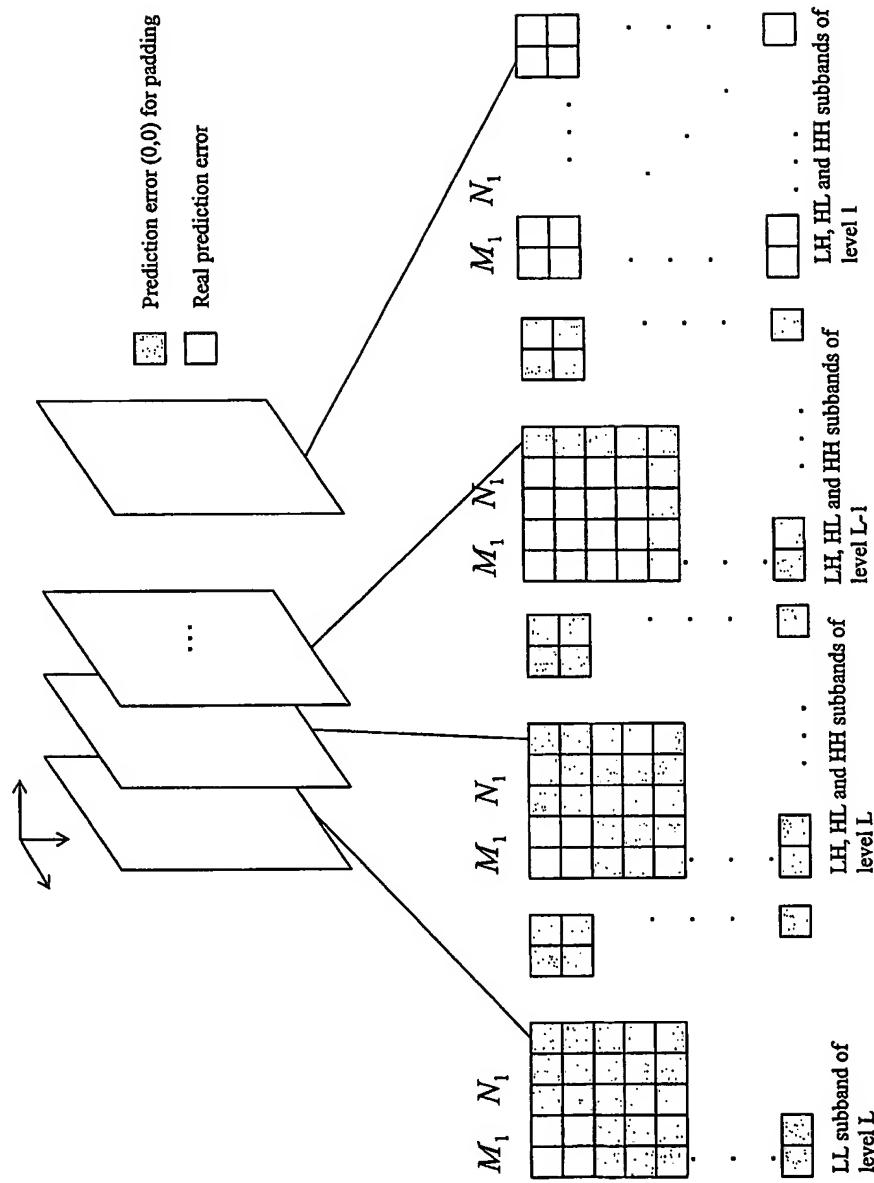


Figure 10

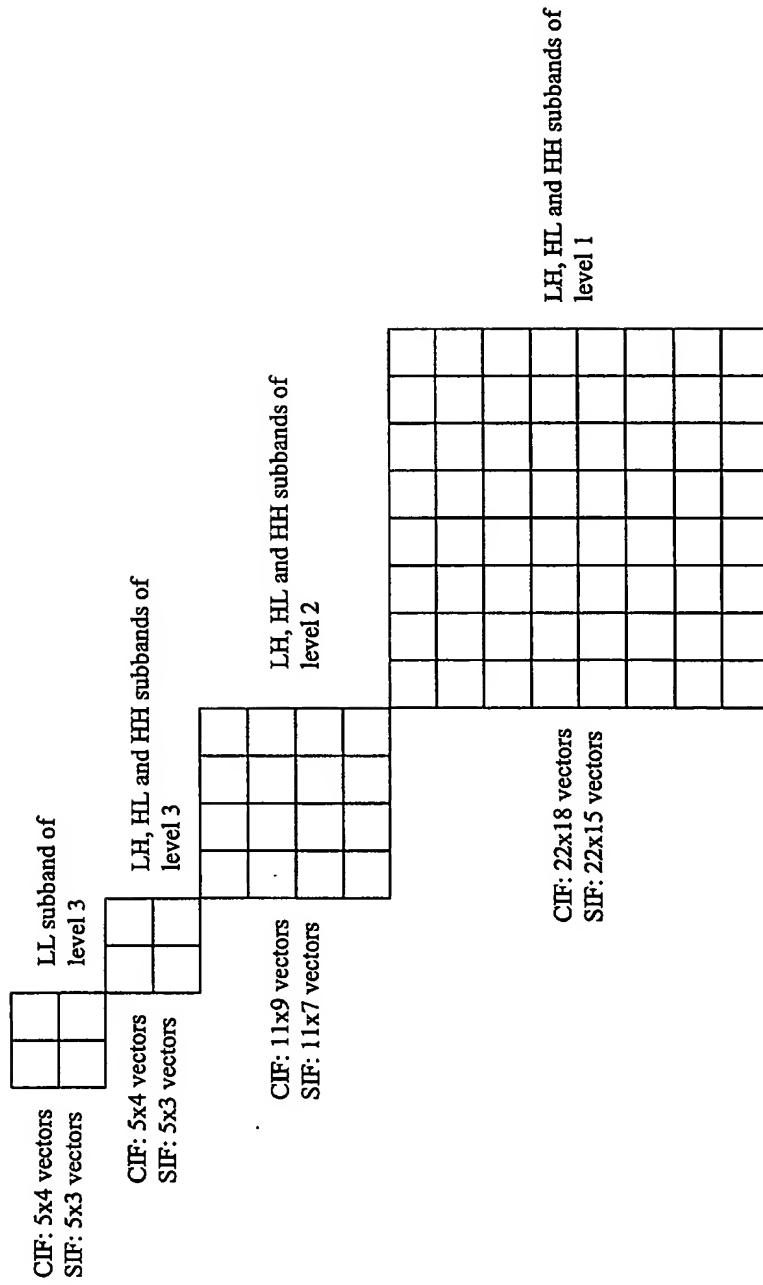
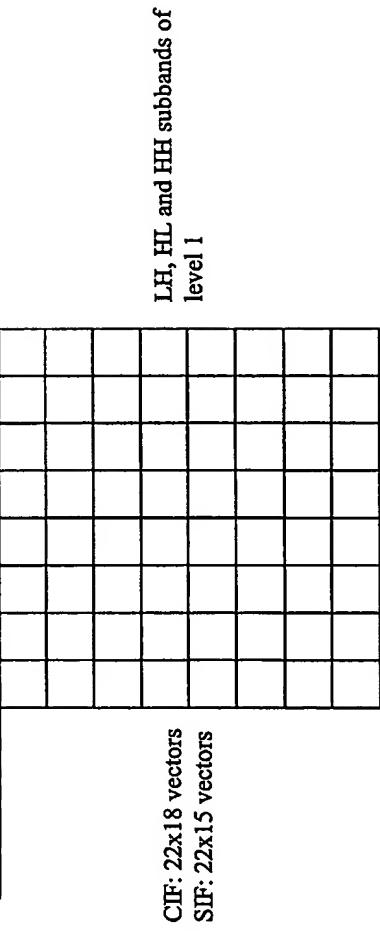


Figure 11



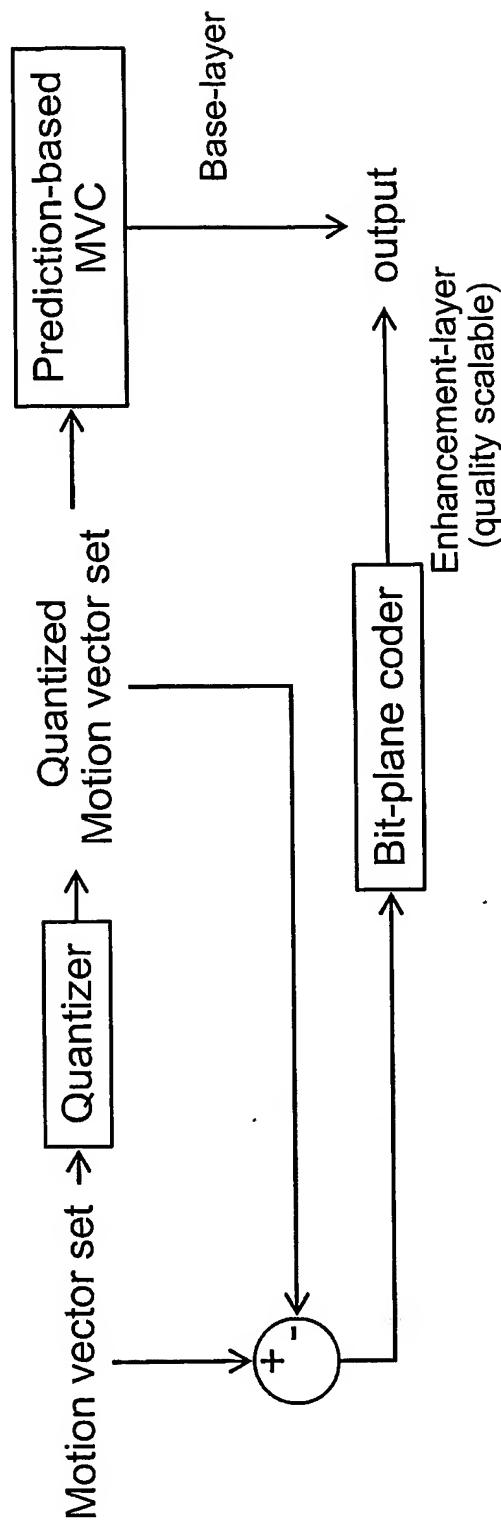


Figure 12 (a)

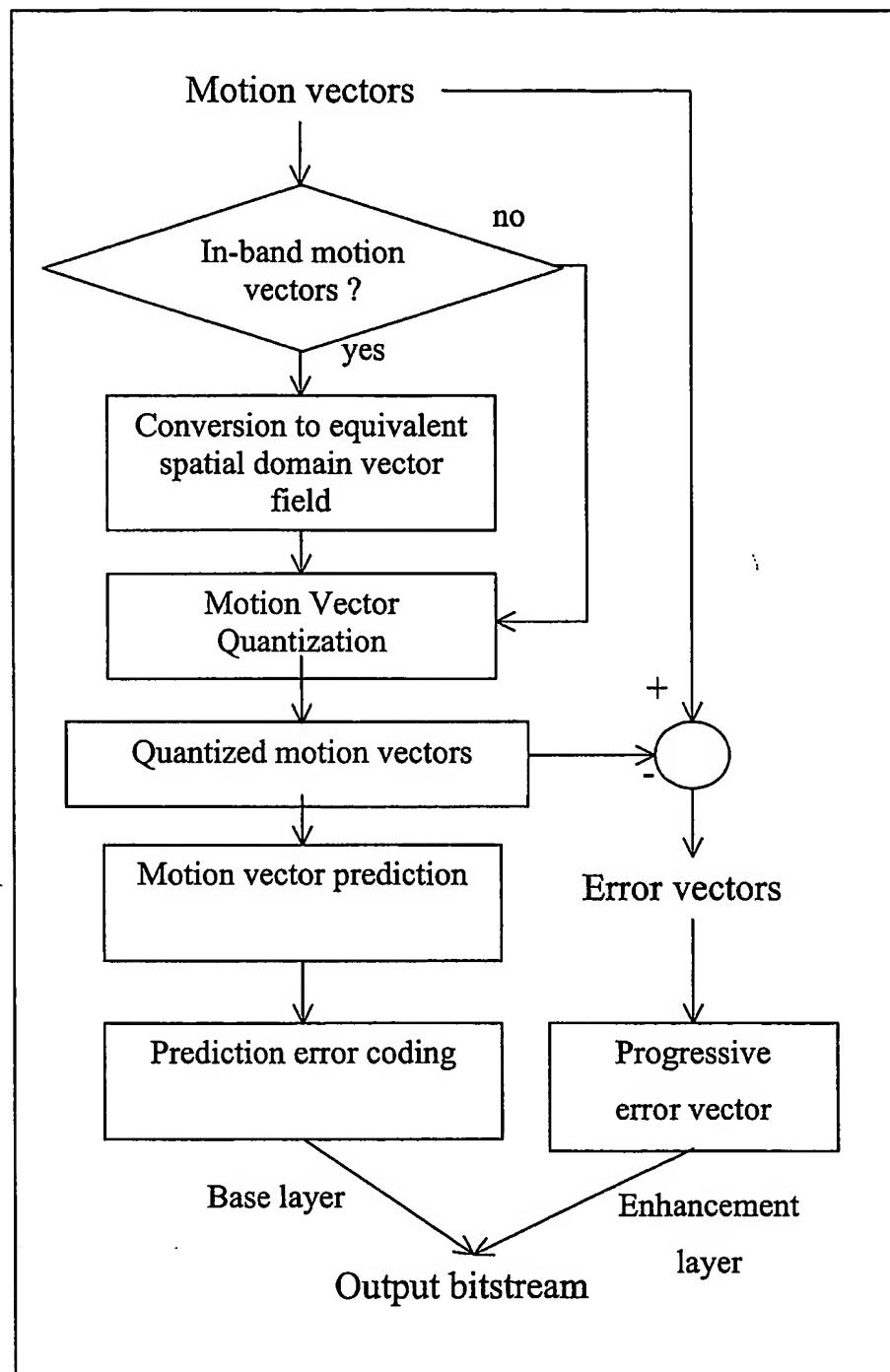


Figure 12 (b)

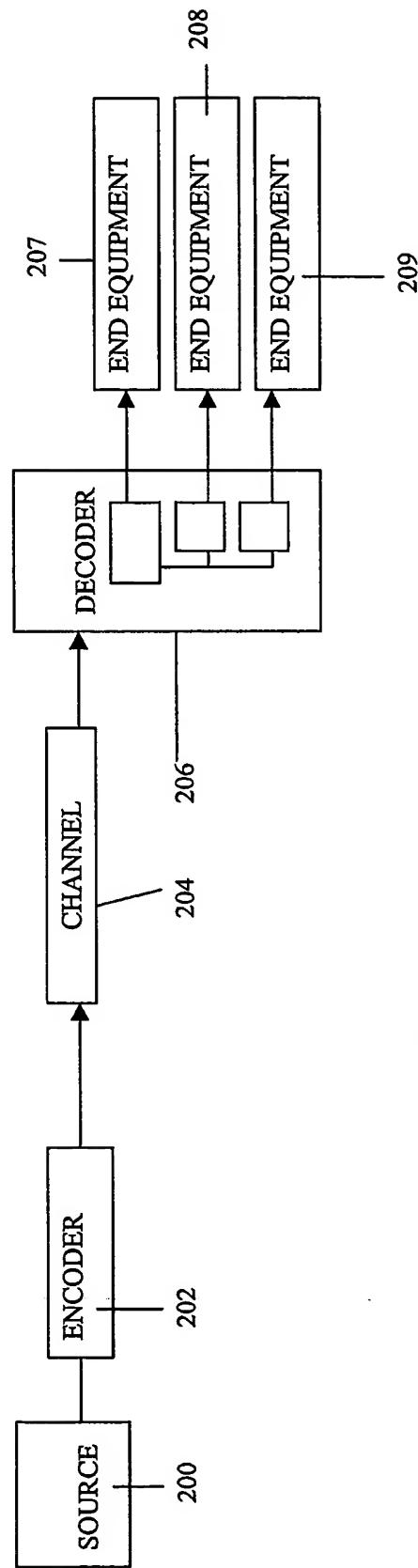


Fig. 13

